

DYNAMIC WWW INTEGRATION SYSTEM

WWW.EASYWEBCREATION.COM

THOMAS FAUQ

**Submitted in partial fulfilment of the requirements of
Napier University for the degree of
Bachelor of Engineering with Honours in
Multimedia Systems**

**School of Computing
May 2002**

Authorship Declaration

I, Thomas FAUQ, confirm that this dissertation and the work presented in it are my own achievement.

1. Where I have consulted the published work of others, it is always clearly attributed.
2. Where I have quoted from the work of others the source is always given. With the exception of such quotations this dissertation is entirely my own work.
3. I have acknowledged all main sources of help.
4. If my research follows on from previous work or is part of a larger collaborative research project I have made clear exactly what was done by others and what I have contributed myself.
5. I have read and understand the penalties associated with plagiarism.

Signed:

Date:

Matriculation Number: 00203424

Abstract

From the early beginning of the World Wide Web known today, there have always been trends toward giving more and more functionalities and services to people who are browsing on this huge network.

This project is about giving the opportunity to newcomers and novices to build a web site online very quickly and easily without any particular knowledge in this domain. Obviously, there are no intention of replacing commercial fully-featured and state of the art software package on the market, but more to focus on novices and newcomers and give them the opportunity to build their own website by the only way they know or have access to: the Web.

The development of <http://www.easywebcreation.com> was done using server-side scripting language PHP associated with a database management system called MySQL. Usability studies gave a constant understanding of the problem that had to be considered.

A web site was achieved and it gives the opportunity to users to register and then create their website in around ten minutes. It is then possible to come back and log again at any time to edit the pages contents. Furthermore, site created will be hosted automatically by *easywebcreation*.

Table of Contents

AUTHORSHIP DECLARATION	2
ABSTRACT	3
TABLE OF CONTENTS.....	4
LIST OF TABLES	6
LIST OF FIGURES	6
ACCOMPANYING MATERIAL.....	7
ACKNOWLEDGMENTS.....	8
1. INTRODUCTION	9
1.1 BACKGROUND.....	9
1.2 AIMS AND OBJECTIVES	10
1.3 REPORT LAYOUT.....	10
2. INVESTIGATIONS.....	12
2.1 INTRODUCTION.....	12
2.2 NEEDS ASSESSMENT	12
2.2.1 <i>Target audience</i>	12
2.2.2 <i>Similar existing services</i>	13
2.2.3 <i>Conclusion</i>	16
2.3 FRONT-END DESIGN ANALYSIS	17
2.3.1 <i>Programming languages analysis</i>	17
2.3.2 <i>Database systems</i>	21
2.3.3 <i>Usability studies</i>	21
2.4 CONCLUSIONS	23
3. DESIGN.....	24
3.1 INTRODUCTION	24
3.2 SYSTEM SPECIFICATIONS	24
3.3 DESIGN OF THE USER INTERFACE	25
3.3.1 <i>Architecture</i>	25
3.3.2 <i>Design rationale</i>	27
3.4 DATABASE DESIGN	28
3.4.1 <i>Architecture</i>	29
3.4.2 <i>MySQL and PHP</i>	29
3.5 CONCLUSION	30
4. IMPLEMENTATION	31
4.1 INTRODUCTION.....	31
4.2 TECHNOLOGIES EMPLOYED	31
4.3 MODULES IMPLEMENTATION	32
4.3.1 <i>General Web Site and login module</i>	32
4.3.2 <i>Management Module</i>	33
4.3.3 <i>Creation Module</i>	36
4.4 DATABASE IMPLEMENTATION	36
4.5 CONCLUSION	36
5. EVALUATIONS.....	37
5.1 INTRODUCTION.....	37
5.2 HEURISTIC EVALUATION	37
5.3 FEATURE INSPECTION.....	37
5.3 ETHNOGRAPHIC RESEARCH	ERROR! BOOKMARK NOT DEFINED.
5.4 SUMMARY OF STUDIES	38

6. CONCLUSION	39
6.1 EVALUATION OF ACHIEVEMENT	39
6.2 DIRECTION FOR FUTURE WORK.....	39
APPENDICES	41
APPENDIX 1: MYSQL DATABASE TABLES ARCHITECTURE	41
APPENDIX 2 :.....	44
APPENDIX 3: PHPMYADMIN INTERFACE	46
REFERENCES.....	47
BIBLIOGRAPHY	48

List of Tables

Table 1: MySQL functioning example (simplified)	29
Table 2: phpMyAdmin features	36
Table U: colors_css MySQL table	41
Table V: config MySQL table	41
Table W: pages_content MySQL table	42
Table X: properties MySQL table	42
Table Y: session MySQL table	42
Table Z: user MySQL table	43

List of Figures

Figure 1: Report Layout focused on investigations	11
Figure 2: Simple HTML File Request	17
Figure 3: Request of a file containing server-side scripting	18
Figure 4: Web Site Structure Chart	25
Figure 5: Creation Process	26
Figure 6: Site manager	27
Figure 7: Storyboard of the web site	28
Figure 8: Colours and CSS editing menu	34
Figure 9: Script Prompt for page content editing	35

Accompanying material

A CD-ROM containing the complete project source code.

Acknowledgments

I would like to thank my project supervisor Dr William Buchanan for the help provided during this project and for the interesting subject and purpose of it.

1. Introduction

With the never-ending growth of the World Wide Web, it often becomes a necessity and a will to show to the world who you are or what your passion is. Unfortunately, it is far easier to browse the web than developing something to push on it, and especially a web site. Many people know the basics of browsing even more than using a computer. That introduces the need of using a web interface instead of a specific and complicated software package to build a web site.

1.1 Background

This project has been born out of a need to provide a different way of building web sites. It is not exactly a brand new idea of course but there will be specific topics that will make, in a sense, this project unique.

Actual software packages devoted to web development are numerous and exist for enough time so that they are well implemented. As such, there are generally full of options in which users lost themselves, especially computing newcomers. Even if they tend to improve constantly and become user-friendlier, novices often refuse to try them. This systematic refusal is due to many factors. Users have often a lack of skills and knowledge concerning software packages because they are missing access to relevant technology and that leads to a lack of confidence that the user can master necessary skills. Indeed, a minimal knowledge of computing, file management and design is generally required to use the software properly. In addition, their cost, for the most famous ones, is another factor that can discourage users to use this kind of techniques to design their web site.

Nowadays, the World Wide Web carries all sort of information in a very mature way. It has grown up from static web pages to an even more dynamic content with years and the trend is now to try and transfer useful functionalities directly onto the web or at least to make the users interact more and more with the web. Thus, it already provide innovative services that people are used to find in their everyday life (music, radio, television) or shopping facilities as well as online transpositions of actual computer software functionalities. According to Ted Schadler [1], Forrester's TechRankings group director, Web Services will certainly be "The Next Technology Thunderstorm" of the web. Of course, his very interesting analysis of the market is more appropriate to businesses but this trend will indisputably spread quickly to the entire web. Actually, it is already possible find a huge number of web sites providing impressive and cutting-edge services like those quoted previously.

Thus, in this case, it is intended to develop a Web Service prototype for creating web sites. As such, it will provide the kind of services given by software packages but online and of course, focusing on the major functionalities only, to stay tuned with the restrictions of the web, the features needed by the target users and their skills to exploit them.

1.2 Aims and Objectives

The overall project objective is the creation of a web-based interface allowing people to create their own web site quickly and without any knowledge in web design. Compared to actual expensive software packages, it allows users to create simple web sites for free and with another level of freedom. All they need to remember is the address of the web site, <http://www.easywebcreation.com>. This way, they do not need to install any piece of software and they do not even need to have a personal computer as soon as they have a standard internet connection. To make it possible, user recognition facilities allow registered users to create and manage their project in several times and in different places. Every bits of their web site in development is stored on the server and in the server's database. Thus, it is possible to edit every elements of the project and update them at will.

The service provided is meant to cover the simple necessity that can occur for someone who wants to put some information online quickly and easily and does not need a highly detailed web site. As such, *Easywebcreation.com* web site, created for the purpose of the project, is very quick, reliable and most of all, easy to use. Most of all, it is clearly stated that the project will focus on inexperienced people, even if every kind of user will be potentially able to find an advantage of using it rather than a more complex one.

More specifically, this project hopes to:

- Provide a review of actual web development commercial software packages and online tools
- Assess the suitability of the different languages and database systems to be used for this particular system
- Develop the web based application using specific methodology
- Develop the database and make it interact with the system
- Evaluate and analyse the resultant application and compare it to existing packages
- Analyse the users interactions with the application and find its weaknesses and strengths

1.3 Report Layout

The report is split into six chapters, including this one and will be laid out as follows to examine the issues stated above. The internal structure of the report was based on the following sketch (Figure 1), reproduction and adaptation of the structure defined by Lee and Owens [2] in his book *Multimedia-Based Instructional Design*.

Chapter 2 is concerned with the investigations that have been necessary at the first stage of the project. It is a major part and will be divided in the two major parts stated on *Figure 1*. First, the project needs, essential analysis to define the future project, will be assessed, and then will be discussed a front-end analysis, composed of investigations on web design tools and languages and then on design techniques.

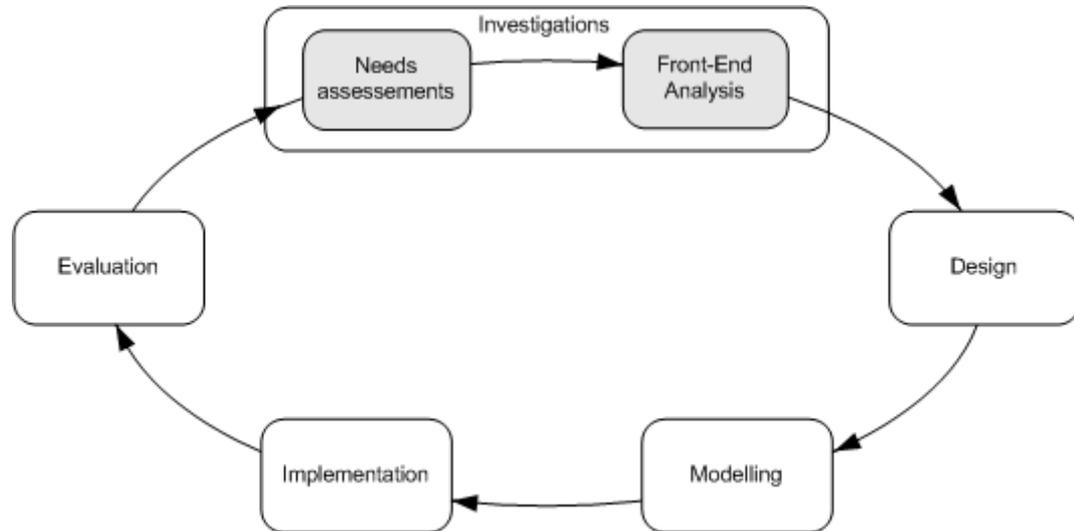


Figure 1: Report Layout focused on investigations
Dick and Carey's Model

Chapter 3 is concerned with the design and modelling of the project. This part will allow us to obtain a better understanding of how the implementation will have to be lead and which consideration will have to be followed.

Chapter 4 details the implementation of the design. As such, the different technologies used will be defined and the most important points will be highlighted.

Chapter 5 investigate the testing and evaluations methods to validate the implementation and its impact on users.

Chapter 6 finalize the report in providing conclusions reached after carrying out the project.

2. Investigations

2.1 Introduction

Before undertaking any kind of concrete development, a deep understanding of the subject is highly essential when a project begins. Developing a Web Service is not a casual thing to do, and it implies to look for knowing perfectly the development environment.

In this particular case, it is obvious that investigations in several fields are needed. They will be divided in two major parts. The needs assessments will give a deeper understanding of the system requirements, the target audience and the existing web development packages or equivalent services. On the other hand, a front-end design analysis will be conducted to investigate the tools and languages needed to establish the project in good condition and thereafter the design techniques to respect. Finally, usability studies will be defined and assessed to see what behaviour must go along with the development of the project.

2.2 Needs assessment

According to Lee and Owens [2], needs assessment is “a systematic way of exploring and establishing the type of solution needed”. Therefore, it is the analysis of the future project by studying what functions and services it must provide and in which way they must be provided.

The project places itself between web development software and online lucrative web services. In all cases, these services are often reserved to firms and people used to computing facilities and programming. To assess and identify what kind of service must really be provided; several investigations must be conducted, including an identification of the target audience, as well as an objective study of existing services similar to the project.

2.2.1 Target audience

An exhaustive study of the target audience gives the opportunity to define the range of people who will be interested by the project and then to adapt it as much as possible to their requirements.

According to the latest survey in date from Nielsen//NetRatings [3], there has been approximately 39 million people using the Internet for the first time in the second half of 2001. In addition to those who already have connection, it shows the potential of people who may want to quickly add information on the web as soon as they have seen its potential. Of course, it would extremely wrong to draw conclusions from numbers but at least it shows that the growth of internet users is constant and then that needs like web site creation will have to be fulfilled.

Furthermore, a good number of internet users have no particular skills to use a computer or a piece of software other than a browser. However, potential target audience must fulfill specific conditions to take advantage of the project.

First of all, they must obviously be able to access the internet. No matter if they have a personal computer with an internet connection or access internet from another place. Moreover, they must have in mind a will to master a need, in this case, creating a website. Then, people who will use the project will be people that have decided to and have looked, by the mean of search engines for instance, for a service that can help them quickly and easily. This kind of attempt requires a certain maturity, which leads to the idea that the majority of users will certainly be adults and teenagers. In addition, recent surveys from National Statistics [4] shows that people who connect in majority are those previously quoted. Anyway, it gives an overview of the majority of people who are susceptible to use the web service but of course it clear that everything will be done not to disadvantage other categories of people.

In terms of skills and knowledge, it was a prerequisite to the project to focus on helping people unacquainted with web development tools. As such, the project will maybe seem not versatile enough to people who are familiar with web development tools or authoring. Actually, it's probably not the kind of service they will look for as they know how to create a web site by themselves. But remembering that it takes a lot of time, it may them a quick way of creating a modest web site for free, what can't be excluded sometimes but it will certainly be very rare.

Then, the category of people potentially interested by the project will be those who don't use the full potential of internet yet because of many reasons, including a lack of skills and knowledge, and a lack of confidence in using new technologies or new services. For those needing to build a website, the project, referenced as an easy tool, must help them overleaping their primary apprehensions.

To conclude, users will be in majority adults and teenagers, from 13 to 55 years old and more who have only basic skills in computing and design. Because of the use of internet, it is also to say that people aimed at have probably a quite good educational level. As stated by the famous analyst Jakob Nielsen [5], the web "continues to be quite an elitist medium", what has to be taken into account. Of course, users will be composed by male and female, and they will only need to be familiar with internet navigation and its most common vocabulary. This is a large potential range of users and an important point for the web site of the project is to have an easy to remember name in the wide range of existing addresses and services of the web. Of course, many usual and short names were already attributed and "Easywebcreation" was finally chosen, for is simplicity and ease of remembering.

2.2.2 Similar existing services

Web design tools became more numerous proportionally to the need of creating web sites and to the growth of the Internet. Since the World Wide Web has been created in 1989 by Tim Berners Lee, there has been a long time since web-authoring tools exist. Then, the project will not of course directly try to compete with state of the art software or other kind of services but rather trying to scrutinize what are their

functionalities and see what can be changed or adapted to the web during the implementation of the design.

- **Web authoring tools**

Web authoring tools gives the opportunity to build a very complete web site with advanced features and style. From personal web page to business web site, the same tools can be used as their functionalities are now levelled out and they can handle without problem the construction of any kind of web site. They are constantly upgraded to take advantage of the latest technologies and in the same time, they all offer more and more similar functionalities.

Depending on the type of authoring tool, knowledge of HTML code is not even needed, in theory, to create consistent sites with impressive features like navigations bars, dynamic HTML, pop-up windows, JavaScript etc...

According to the users' level of expertise, web-authoring tools provide different functionalities. They can be divided in three categories:

- Code-based editors
- WYSIWYG (What You See Is What You Get) editors
- Mixed editors

Code-based editors are what were used at the beginning of web pages creation. First pages contained only text and hyperlinks so that of course nothing else than a text editor was really needed. It has now evolved in pure improved code-based editor assisting users in their tasks by helping in coding raw HTML, set options and organizes the general layout of the page. Basically, common features are colour-coded source, code completion, spell checking or cascading style sheets editing. They do not add unwanted code as user keeps control of everything. This kind of editors are really designed for developers or at least people who know HTML code very well, and even more additional languages like client-side (JavaScript, VBScript) or server-side (PHP,ASP,JSP...) scripting languages.

One of the best code-based editors is Macromedia HomeSite. TextPad or WebExpert are other examples of poorer but very useful code-based editors.

WYSIWYG editors are editors entirely based on the visual appearance of the web site. They let the user make them without the trouble of hand coding HTML. It acts, more or less, as if a publishing program were used. It is the easiest type of web authoring tool to use, as you know exactly during the whole development process how your site will look like. In the same time, you need to show creativity and ability to design the web site from the beginning. If templates are provided, the task will either been much facilitated. However, there are of course inconvenient, and WYSIWYG editors have often been criticized for their terrible lack of capacities in generating good HTML code and their difficult to use interface. NetObjects Fusion is probably the most famous WYSIWYG only editor, and it has exactly the characteristics quoted previously.

Mixed editors combine the two previous systems to provide a complete web design solution. Macromedia DreamWeaver is by far the most famous and the most complete

mixed editor. At the same time, it is so complete that it is squarely aimed at the web developers more than anybody else. As such, it provides a strong panel of options, integration of cutting-edge technologies and languages and a very good way of handling code without adding proprietary code. That is for instance different from Microsoft FrontPage which adds a lot of proprietary code and which reputation is not very good among professionals. FrontPage is more to be used by users who got the advantage of finding it in their Microsoft Office package.

Each type of web-authoring tools has its strengths and weaknesses. Code-editors are useful but reserved to experts, WYSIWYG programs are supposed to be very easy to use but produce terrible HTML code and mixed tools seem to be more dedicated to professionals. Anyway, as mentioned in the Business Week Online [6], "In addition to the cost [...], you can expect to spend some time learning to use the program effectively". This reference quotes the two major issues that remain with these software. First of all, they cost a lot of money, from around £65 for code-based editors to £299 for an impressive tool like Macromedia DreamWeaver. It is a difficult investment to take into account if the goal is merely to design an honest web site quickly. On the other hand, even when very well designed, web-authoring tools, because they offer so many functionalities, need users to spend some time learning them, and it is often the opposite of what the kind of pre-defined target audience is looking for.

- **Open Source reusable components** (Content Management Systems)

Another way of creating a web site is to use Content Management Systems (CMS). They are generally freeware and dedicated to a very specific task. Most of them indeed are dedicated to documents or publishing content management. Actually, these hybrid solutions provide essential administration online but require a previous specific configuration and installation. They are generally based on PHP/MySQL and their primary purpose is to allow people to maintain dynamic content of a web site quickly and in a reliable manner.

PHP-Nuke, recognized as the best CMS available, offers an impressive set of functionalities. User login, themes, web links or additional modules are part of the multiple options available. And because it is written in PHP, it's even entirely customizable by hand to cope with your exact needs. But this is also the limitation of such systems, because skills are really needed to implement such systems, and most of all, it only covers news or publishing web sites creation.

Content Management Systems are really helpful for many news web site administrators who have seen their work eased by far with such systems. There are online administration facilities and even registered users can contribute to the web site so that, in a sense, the web site can finally have state of the art content without any actions coming from the administrator. This is exactly what is meant by dynamic content and as such, CMS gives to users functionalities that are generally available in much more expensive and professional packages.

- **Web-based services**

Web-based services are the kind of project involved here. There are not numerous yet but it is possible to find out that great ones already exist. As aforementioned, the trend is now in developing new web services and web site creation systems do not flee the rule.

After deep investigations, it became obvious that it was difficult not to say impossible to find independent and free web-based services, except from old project that have not evolved for months. Commonly, Web-based services place themselves in two major categories.

One of them is lucrative web creation service that claims a certain fixed amount of money or a monthly charge for the service provided. Jpowered [7] purpose to let you create your web site with great services and web templates choice from \$197 (~£138) to \$397 (~£277). ZyWeb [8], with the same kind of professional service, claim £5.50 a month. This is quite a lot of money but they provide professional looking websites and it is really worth the price.

The second kind of service is more or less identical but free and basically lighter in term of functionalities. In fact, it is really the same kind of service that will be provided by the project implementation but it is part of an entire system and dependant of it. In that, these services are only available for people as an additional service, and they won't be able to use their web site elsewhere than hosted by the same company. Moreover, they have no control over it and cannot even save their data. This valuable service can be found with Lycos.com that propose a specific web builder as soon as you've created an account and ask for help in creating a web site quickly and easily. It works great and is probably meant to offer valuable services to clients for free to create a customers base that will be useful when becoming payable.

With existing web services, there is either a problem of excessive cost or independence of services provided. The majority of novices and beginners won't probably pay a big amount of money for a basic and personal web site. In the same time, they must have specificities making them unable to use proprietary services that oblige them to keep their original settings, as their free host original service for instance.

2.2.3 Conclusion

This need assessment has been a good solution to study which kind of audience the project must be focused on. It gave as well a better understanding of the existing market of web design software and solutions in terms of authoring to find out what could make the difference, and help the target users.

It seems obvious now that from a solution to another, the level of complexity, features and price is varying a lot. The project will try to make the difference by finding the right solution to adapt for the target audience.

First of all, the trend will be to really focus on novices and beginners, principally adults, and try not to discourage them to create their own project. In the same time, it is obvious that goals cannot be to replace professional services that have cost thousands of pounds to develop. At the contrary, Easywebcreation project will try to take advantage of the gap between what exists, complex and expensive, and what will be realized, simple, honest and free. Of course, to develop such a service, much more information and investigations on the kind of system to design are needed and that is what will be studied later on.

2.3 Front-end design analysis

Web design is an exiting and noble undertaking, but one that is fraught with pitfalls. Then, this is fundamental to study the several techniques and tools available to develop the system before beginning any kind of design or implementation. The purpose of this analysis is to define which programming languages and database systems will be the most appropriate to the development of the project and why. In the same time, investigations on usability will allow us to know how to behave during the design and implementation of the system.

2.3.1 Programming languages analysis

To provide a dynamic integration system, powerful tools are needed. Many web sites are composed of static content only, but as soon as more interaction is required, dynamic content is necessary. In order to generate web pages dynamically, servers need to execute a script. Then, a specific server-side scripting mechanism will execute a program that will interpret it and return as a result a formatted HTML page. To do so, the web service has obviously to be developed using server-side scripting technology. Actually, it has the advantage over client-side scripting in being totally transparent to the client and independent of its configuration. Then, it will run on any platform with any browser able to run HTML. It is also far more powerful and increase security as the original code is not viewable by the user because of the interpretation.

Here are sketch explaining the difference between static and dynamic pages:

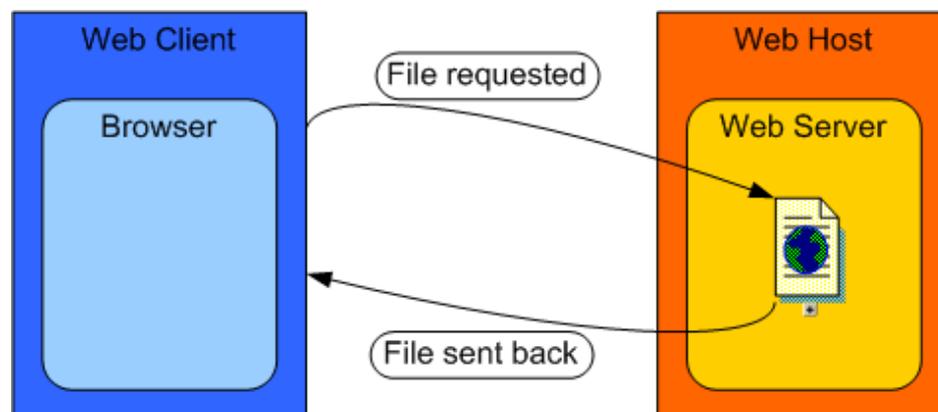


Figure 2: Simple HTML File Request

On Figure 2, a basic HTML file request is shown. The file is stored on the server and downloaded by the client when requested. Then the browser interprets the HTML content and displays the result on screen.

Figure 3 below shows an example of what happens with pages including server-side scripting:

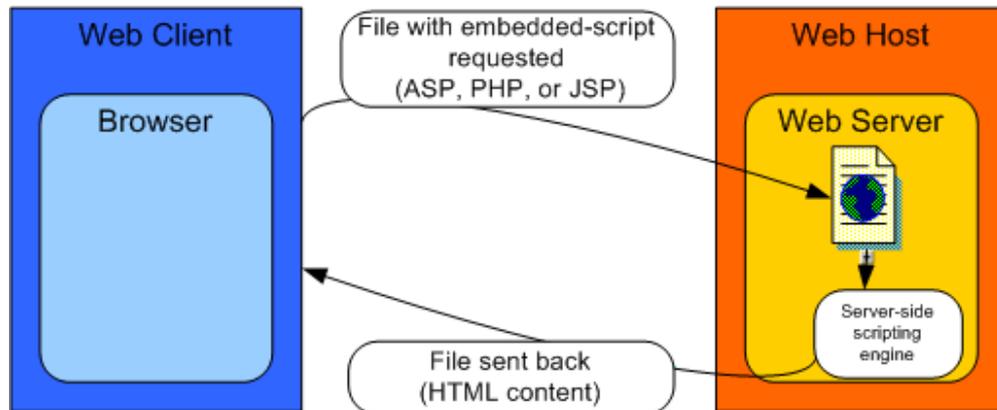


Figure 3: Request of a file containing server-side scripting

In the situation shown in Figure 3, a file is requested and its extension tells the server that he has to interpret it. Then, it parses its content thanks to the corresponding server-side scripting engine and return basic HTML file that has been automatically generated.

Subsequently the most actual and reputed scripting technologies will be investigated to define the more suitable to the project. First technology to be released was CGI (Common Gateway Interface) Scripts which has nowadays some performance shortcoming. To cope with those lacks of performance and replace CGI scripts, other technologies have progressively emerged. These technologies embed the scripts directly into the HTML page instead of using a separate file containing the script. Microsoft Active Server Pages (ASP) is one of them, and supports a wide range of databases systems. Java Server Pages (JSP) is Sun alternative to server-side scripting and can be used in conjunction with Java Servlets. Finally, PHP (PHP Hypertext Pre-Processor) is the third big alternative to CGI scripting. It is free and open-source; which provide instantly a great success to this technology, in addition to its multiple functionalities.

- CGI Scripts

CGI Scripts, often written in PERL (Practical Extraction and Report Language), are executable compiled components (.exe or .dll files) that produce HTML content on the server that will be sent back to the user. CGI is the first server-side technology to have had a real popular success over the internet. It gave the opportunity to add interactivity to the web, thanks to functionalities like writing in files, access databases or create HTML dynamically. CGI programs are executed in real-time, which means than they are very fast. Nonetheless, they are highly dependant of the platform that

runs the scripts and then can see the performance decrease sometimes. In addition, CGI scripts are run in separate processes each time they are requested, and this lack to handle concurrent client requests can easily create huge overhead on the server.

CGI are less and less used nowadays due this kind of issues and the release of new systems, easier to use and more powerful. Moreover, CGI is reserved to a certain category of programmers as it is essentially composed of PERL or C/C++, which are very complicated programming languages.

- ASP - Active Server Pages

ASP is Microsoft's solution to server-side scripting technologies. It has been developed as an attempt to solve the inefficiencies of CGI scripts and the difficulties of developing them. It is based on scripts embedded in the HTML page. Scripts are written in VBScript, Jscript or JavaScript. As aforementioned, this scripted are interpreted by the server which send back to the client a standard HTML page.

ASP has the advantage to be easy to implement compared to CGI. Thus, it is possible create interactive and dynamic ASP pages without being a professional programmer. Collect information form a submitted form, personalize a HTML document depending on the user or the detected browser are functions that can be easily implemented without much knowledge in ASP programming. It is not necessary to know the entire language to implement such functions, which saves lot of hours and really simplify development. With a better understanding or a need to accomplish much better scripts, ActiveX components can be used. This is for instance necessary to access a database.

Largely spread on the web, ASP technology drawback is that it works nearly exclusively on Windows platforms running Microsoft Internet Information Servers (IIS). ActiveX being Microsoft exclusive technology as well, it seems obvious that people not using Microsoft tools will not be able to take advantage of that kind of solutions.

- JSP - JavaServer Pages

JavaServer Pages is a server-side scripting technology based on Java components and developed by Sun Microsystems. In some respect, it is quite similar to ASP or PHP. It distinguishes himself from other scripting languages by being extensible and having no ownership. Being an extension of Java Servlets, it typically benefits from the advantages of Java. It has the reputation of being very fast, platform independent and highly scalable. Most of all, it is generally free and compatible with most web servers.

With JSP, the code can be embedded in HTML pages as well. In this case, JSP is integrated with XML-like tags. JSP is the younger server-side scripting technology available and seems to offer powerful functionalities and robustness. Nevertheless, as it takes advantages of Java language, which is not particularly reputed for its simplicity, developer must be used to the language. On another hand, facilities to implement JSP are still rare, especially for non-professional users.

- PHP - PHP HyperText Preprocessor

PHP is a server-side scripting language very similar to ASP and JSP. As them, it is embedded in HTML files and used to create dynamic web pages.

First created for personal use, its open-source characteristics have led it to a much more efficient, reliable and faster language. PHP is entirely free and the fact that it is open source makes the language improve at much quicker rate that firms can even conceive with traditional development. Indeed, UNIX inspired attributes have made of PHP an impressively successful language becoming the “de facto way of developing web applications” in the Linux environment, according to a Netcraft survey [9].

Totally created for the web, PHP includes functionalities that are really trivial to implement. Thus, e-mail sending features, image or graphics generation on-the-fly or creation of PDF documents are some of the impressive immediate features of PHP. It has also the ability to connect to nearly every kind of databases, and is generally combined with MySQL databases which provide an ideal and simple solution.

PHP has been created out of a need from web developers and as an open-source project, has evolved in the way people wanted. That is surely the reason of its huge success.

- Comparison and conclusion

This is not to say, server-side scripting languages propose all very similar functionalities. Except from CGI scripts which are a bit more complicated to implement and show weaknesses in term of handling lots of executions, ASP, JSP and PHP are all embedded in the HTML code, then parsed and interpreted by a specific engine. Then, more than focusing on their features, it is important to select a specific server-side programming language in terms of the development environment. To use such systems, an access to specific technologies is needed, especially the need of a server access. In the same time, the solution must be chosen in terms of programming skills, and also in terms of the audience aimed at.

ASP is very powerful language but has the inconvenient to run exclusively on Microsoft platform. Furthermore, the use of activeX to implement better functionalities reduces a lot more the panel of users. In addition to that, it is a proprietary product and as such, there is incertitude in terms of how the language will evolve.

CGI and JSP scripts have a reputation of being complex to implement. While CGI shows weakness, JSP, though it is powerful and reliable, runs on most servers but they still have to be implemented to accept it, which is rare in non-professional environment.

Then comes PHP, which is open-source, very well documented and can be used on most web-server environments. Its popularity makes hosting easy to find and it is reputed for being the most accessible of all the aforementioned server-side scripting languages. Besides, free web hosting generally propose by default MySQL database

system, which simplifies the global vision of the project a lot. That is why it seems to be the most reasonable choice for this kind of project.

2.3.2 Database systems

Dynamic web site design implies generally the use of a solid database to add functionalities and improve the service provided. In this case, the database system will be a major preoccupation, as the entire system will be based on it. As PHP was previously chosen as the server-side scripting language to develop the web service, and even if it can handle nearly all kind of databases, there are special facts to take in considerations because the project will be hosted by a free web host and they generally only provide MySQL services.

MySQL is the system of choice for many webmasters, and it is one of the broadest database system associated to PHP on the internet. It is purely open source project and licensed under the GNU General Public License (GPL). For two consecutive years, it has even received the award of best database system in Linux environment by Linux Magazine journalists [10]. Say, MySQL is fast, provide excellent functionalities and is supported by wide range of API (Application Program Interface) and clients. In fact, this is generally the database system recommenced when looking for a solid, stable, and feature-rich database environment.

Actually, alongside the fact that there is not really a wide range of possibilities in term of choice, MySQL properties are largely sufficient and impressive to conclude that it will be a perfect database system to use with PHP for this project. The fact that free hosts generally provide a specific API called *PHPMyAdmin* [11] to use MySQL is another argument which leads to this choice, because it really ease the administration of such a database.

2.3.3 Usability studies

Usability methods are essential investigations in the process of project creation. It is very important indeed to define how each step undertaken will be evaluated and laid open to criticisms. Then, during the whole process, it is always essential to make sure the project is easy to use, meets requirements and objectives. The developer is not capable of investing this kind of issues alone because of his too deep relation and knowledge of the project. As it is said in *A practical guide to usability testing* [12], “users decide when a product is easy to use”. That is why it is necessary to follow predefined methods concerning usability, it will help the designer to develop adapted systems and evaluate them. The methods that seems the most useful to this particular project are defined below.

- Heuristic evaluation

Heuristic evaluation is a method consisting of a systematic inspection of the user interface design for usability. It is conducted during the entire project and as such, is really helpful and valuable. By evaluating the system regularly, it is made possible to unmask hidden usability problems and solve them before the project deadline.

Heuristic cannot be driven by the only designer because all usability problems cannot be found by only one person. Three evaluators seem to be the minimal number for a good evaluation. Evaluators are not exactly real users as they need to have a minimal knowledge of usability to find the system weaknesses. With them, it will be intended to focus on major issues, like some of the ten major heuristics defined by Jakob Nielsen [13]:

- Visibility of system status
- User control and freedom
- Consistency and standards
- Error prevention
- Recognition rather than recall
- Flexibility and efficiency of use
- Aesthetic and minimalist design
- Help users recognize, diagnose, and recover from errors

Heuristics evaluation can be conducted at each stages of the project management and provide a good understanding of the user point of view and of the usability problems.

- Ethnographic research

Ethnographic researches are evaluations totally focused on the user. Actually, with this approach, users will be evaluated while using the system. The designer will either do an inquiry to study the user behaviour and ask open-ended questions, or conduct a more formal interview while the user is using the system.

This is a great and rich experience, as it makes possible to see the user interacting with the system. As such, the project must be completed or nearly, otherwise the user will not have a characteristic behaviour and the evaluation will be misrepresented. Then this kind of evaluation will happen in the final evaluation phase of the project. It may help identifying usability issues not previously envisaged and avoid the creation of a final product that is far from user expectation.

- Feature inspection

Another evaluation which is not too much complicated to realize, even if volunteers are still needed, is features inspection. What is meant with this method is to test and evaluate if the features provided by the designer are well used by the user. To do so, a little scenario has to be created and submitted to the user, who will assess the usability to realise tasks at each steps. Then it will be possible to define the availability, understandability, and general usefulness of each of them.

This evaluation is ideal in the middle stage of the project development. Actually, what is meant to point at here is not the entire site usability but more to discover how hard each feature is to use.

- Questionnaires
-

Questionnaires are probably the method that is the most know around. It differs from previous methods by the fact that they are written list of questions, not ad hoc interviews or discussions. It requires much more efforts from the users as questionnaires miss the human contact and then become a little bit duller.

The only thing to remember for conducting questionnaires is that questions have to be asked in a way to obtain the wanted answer. Actually, as defined by Human Factor Research Group [14], there can be of two different types of questionnaires.

- Close-ended questionnaires provide pre-written answers and they are created to obtain meaningful numeric data quickly.
- Open-ended questionnaires request the user to answer questions in their own words.

These are two different kinds of questionnaires and can be mixed together for specific purposes.

2.4 Conclusions

Many investigations have been before taking care of considerations that are more practical. It was absolutely necessary as it helped in the understanding of the problem and outlined main specifications to define.

Defining the target audience will be very important in the definition of specification, as it will be mainly focused on novices and beginners, who have not exactly the same needs as other skills users.

Studying the services provided by competitors allowed to have a better understanding of what the project will have to do, and identify in these professional packages what was usable and what was not. The goal of this project is not to give blind conclusions based on utopia. The main goal is not to be better than professional packages developed for years by professional programmers but to define a need and study what could make the difference to help users finding it easier to realize a specific task, which is here to build an honest web site quickly and easily. It is now sure that the project will mainly attend to help novices and beginners in creating a web site and give them the confidence they cannot find in complicated software.

The practical solution needed has been well documented and it was decided that open-source server-side scripting language and database systems were the more appropriate to the project, giving total freedom in terms of hosting and front-end use by the user, and providing great functionalities with less complicated syntax than other languages.

Finally, different evaluation solutions have been investigated and that will help raising concerns about the usability of the system. It has been stated that there were methods to use in all steps of the development, and that they will all help in finding the strengths and weaknesses of the project.

With these investigations conclusions, the process of design can now be considered.

3. Design

3.1 Introduction

Now that all the information needed have been documented in previous investigations, the design phase will help focusing on what is required by the application and how it will have to be implemented.

This section is all about accurately defining how the implementation will be conducted and what pre-requisites must be chosen.

3.2 System specifications

The web service that is going to be developed is aimed at novices and beginners who do not know anything about web design. Furthermore, they are rejecting actual solutions, including state of the art web authoring software. Whether it is because of their lack of confidence, or because they do not want to waste their time in learning how to use such or such system, it is clear that the goal is to try to change that kind of behaviour by providing a simplified system directly on the web.

Previous analysis provided the conclusion that the best solution for that kind of dynamic service was a server-side scripting language associated with powerful database system. It has been seen more in details that the best choice was to use PHP associated with MySQL database.

PHP will be used to generate the web service pages from reusable components and make the content of the site dynamic and responding dynamically to user actions. It will also send commands or requests to MySQL database, where information concerning users, login, password and preferences will be stored and accessible at any time.

The whole system will be based on user recognition thanks to a session number stored in the database. This random created number, always corresponding to a specific user, will be passed to every requested page and will then allow secure user recognition, essential to the good functioning of the different functions.

Thus, users will access the web service online and will be able to create their website quickly and easily. The target audience must necessarily have basic browsing skills and know how to use a mouse and a keyboard. Furthermore, they must have minimal internet knowledge to be familiar with the simple but technical terms used.

In terms of user system requirements, the needs are very poor. A browser and an internet connection are basically the only thing that the user will need on his computer to be able to use the service. In terms of browsers, Internet Explorer will be preferred to alternative browsers because there are sometimes differences of interpretation. Anyway, the service will work as well on alternative browsers, which in all case, represents in 2002 only 7 to 9 percent of the entire worldwide browsers use [15].

3.3 Design of the user interface

3.3.1 Architecture

It must already be anticipated that the web site will be principally a way of welcoming people and tell that what the project is about. As such, it will provide information and links to relevant information. But most of all, it will be the location where users will be able to login or register, and then to decide to create their website if that's the first time they come on the site, or to manage it if they are coming back from another session.

The web site will be divided in three main parts, or modules. The first one will be dedicated to be the portal welcoming people, giving them information and ability to log on and register. The second one will handle the creation process, step after step, to create a website or at least a draft of it quickly and with guidance. Finally, the third module will be the managing system, or "site manager", which will be the location where users will be able to personalize their website and keep it up to date.

- Portal

This is the main part of the web site, the one people will access when typing the URL <http://www.easywebcreation.com>. It will act as an informative and welcoming tool, allowing people to discover what the subject and role of the web site. Furthermore, it is the artifact that will allow access to other modules and as such, it is the guarantor that it is always possible to access them. Here is the general structure chart of the portal configuration:

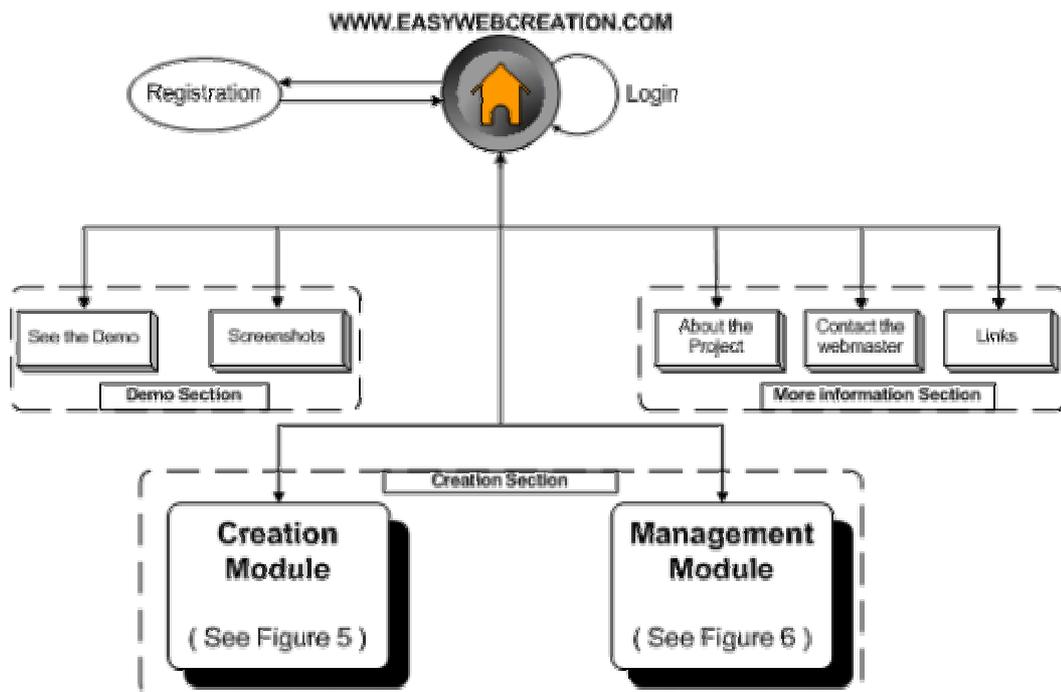


Figure 4: Web Site Structure Chart

As shows Figure 4, this is quite a simple structure because its purpose is very simple. The portal is not the module containing the most complicated parts of the project, but is nonetheless essential for its cohesion. Moreover, in addition to links to other pages, it will provide registration facility and possibility of login. And when a user will have taken the opportunity to register, access to creation page and management module will be available.

- Creation module

The creation module is going to be the first real added-value service provided to the user. It has to provide a simple and clear interface that will guide the user to each steps of the creation. Keeping in mind that the user will probably be misdirected if functionalities and options go too far from their expectations, creation process will be straightforward and as quick as possible. As such, creation process will propose six steps guiding users in the creation of their web site. Moreover, information will not necessarily have to be completed, as the management module can always modify data later on.

The Creation is very linear and can be symbolized like that:

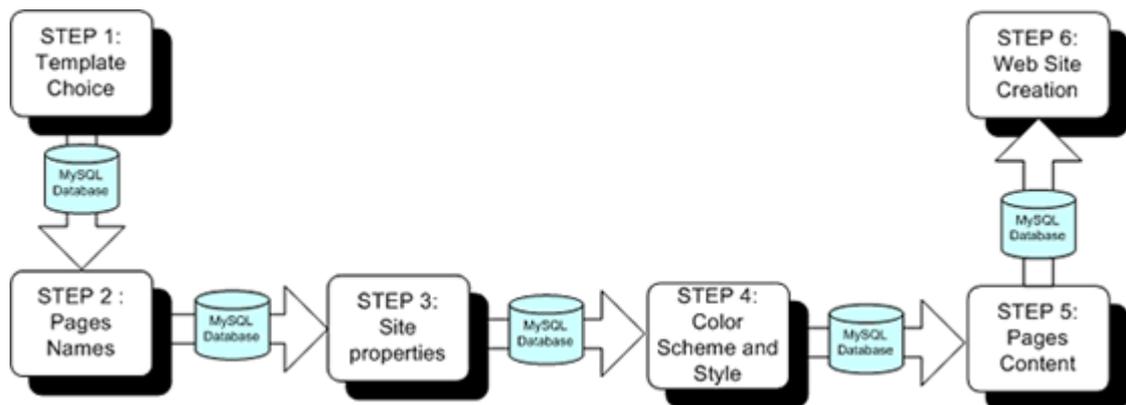


Figure 5: Creation Process

Information will be added to a specific part of the database at each step of the creation so that the final step of creating physically the pages will take less time to execute. When this process is taken to the end, users will typically have nothing else to do that note their new web site address, which will already be usable as an alias of the website defined by their login name as follow:

<http://www.easywebcreation.com/users/theirname>

It will be a good occasion to see the result of their choice and as a first choice is rarely perfect, they will be of course able to edit their properties and choices in the management section.

- Management and editing module
-

As part of a creation process, there is always a need to go back to the decisions taken and change the choice made if necessary. Then, the creation module by itself fulfills perfectly its aim but is not flexible at all, merely because this is not what is wanted. Nevertheless, users may need to change their data, because of errors introduced in the creation process or just later when updates will be needed. This time, instead of guiding user across steps, options are provided to let the user choose what has to be changed:

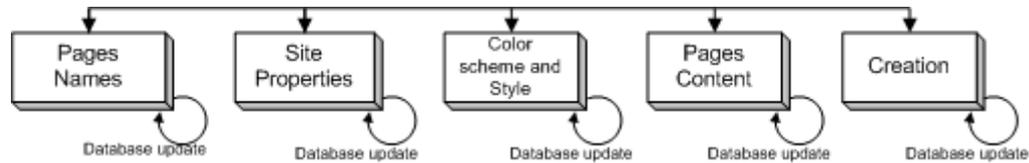


Figure 6: Site manager

Figure 6 shows the non-linearity of the management module. A top menu will provide access to each page wherever the user is and will always be accessible on top of the page. This way, if the user only wants to update information in two sections, he will be able to do that without looking after others.

3.3.2 Design rationale

After considerations on the architecture of the project web site, a brief look at the way information will be visually presented is needed. This process is typically essential to respect rules that make a web site attractive and user-friendly.

Technical literature like *Planning and Designing Effective Websites* and many others quoted in the bibliography illustrate the way of bringing the concept to life and create a simple mental model for the user. Content was posted in a manner that assists users in the navigation and in the use of the site.

It was particularly important to focus on the way the site will be displayed on screen, its speed of loading and its rationality.

With the disparity of actual displays, it is important to respect common rules adopted by professional in term of resolution. Thus, the site will be designed to be fully displayed on an 800*600 resolution, which is the minimal resolution generally observed nowadays.

To keep it rational, the general design will always look more or less the same (See Figure 7 following), and the changing content will be encapsulated in it. As such, it will be composed of a title bar, which will remind the user, very simply, the name of the website actually browsed. A little bar will be placed below it to display additional information. For instance, the date will probably be displayed, as well as information about the navigation or error messages. Menus will surround a central changeable content on both lateral sides to provide navigation functions. On the left, the menu will give access to other categories via links and will always be visible, on all pages. On the right, menu will be especially designed to enter login information. Finally, to

complete the design, a small footer is present at the bottom of each page. It may provide additional information of small importance.

To be sure that these different parts of the website will always be exactly identical, they will be created as modules that will be dynamically loaded on each pages. Thus, the only part that will have to be coded on each page is the adjustable content that can be seen on the following storyboard.

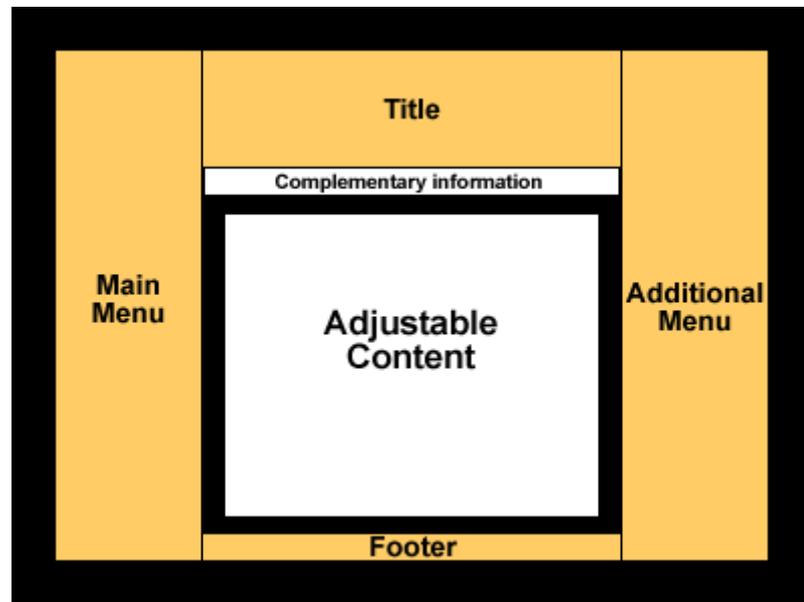


Figure 7: Storyboard of the web site

The only part of the website that will not respect this design rationale completely is the site manager. Actually, two much links and information have to be provided and it will be loaded in a separate page with a different and more appropriate menu.

In terms of colours, those shown on the storyboard will globally be kept in the final colour scheme. Well-contrasted colours for the menu and the background give a feeling of clearness and warmth with a bit of originality. The adjustable content will generally be white to give a high-readability to this part where important information will be displayed.

This web design argumentations gave enough information to be able to develop a logical and functional website. Most of all, it is very important not to have a too poor website design and to show that great design can be developed. This is important as this is one of the argument on which users will point at to determine if they will use the service or not.

Being dynamic, visual design is not the only requirement on which the implementation must be based. Actually, nearly all the functionalities offered to the users will be achievable thanks to the database. Now, the premises that need to be respected in the construction of a database will be investigated.

3.4 Database design

The whole creation process, as well as the site management process, is based on a database system. It has been decided during the investigations that the database system chosen here will be based on MySQL. The following sub-sections will show what architecture must be respected and implemented in a global vision of the system.

3.4.1 Architecture

MySQL is a multi-user and multi-threaded SQL (Structured Query Language) database management system working under Microsoft Windows and Linux.

It has a reputation of being fast, reliable, and easy to use. In the same time, databases systems are the kind of systems that requires seriousness and rigor, and a design without errors. It has to be defined before any kind of implementation and if possible definitively. Requests to MySQL databases are so numerous and so error-sensitive that a simple field added in a table can breakdown an entire system.

Thanks to previous analysis, what the web site will be composed of has been defined and as such, each table to create in the database can be defined as well.

Actually, there will be six tables included in a database called "easywebcreation". These six tables, called *color_css*, *config*, *pages_content*, *properties*, *session* and *user* are detailed in *Appendix 1*.

A special effort to split data into logical parts has been made. Hence, with the specifications included in the Appendix 1, the database will now be very easy to implement.

3.4.2 MySQL and PHP

MySQL databases are often used in parallel with PHP to develop dynamic site and to ease updates and management. As most databases systems, MySQL databases are formed of different tables containing themselves several fields.

The functions to access and maintain databases are really trivial in PHP and allow performing MySQL tasks very easily.

Here is a basic framework of a PHP-driven MySQL transaction (simplified) for a better understanding of its functioning:

1. Open Connection	<code>mysql_connect(host,user,pass);</code>
2. Select Database	<code>mysql_select_db(database_name);</code>
3. Query Database	<code>\$result=mysql_query("SELECT * FROM database_table");</code>
4. Read Data	<code>mysql_result(\$result,0,"field_wanted");</code>
5. Close connection	<code>mysql_close();</code>

Table 1: MySQL functioning example (simplified)

In addition to that, specific commands (INSERT, UPDATE, and DELETE) allow respectively adding, modifying or deleting information on the database. There are of course other functionalities but most of the requests can be done when knowing how to handle the aforementioned command.

3.5 Conclusion

This Chapter has covered and demonstrated what will be the choices for the implementation and how they will be implemented. Indeed, the system specifications have been defined and showed that the web site should remain simple and use PHP/MySQL as practical solution. Most of all, deep studies have been conducted about the website design and its functionalities and how the database must be built. The implementation reserves hard work and the fact that the structure and functioning of the system have already been defined will be of great help.

4. Implementation

4.1 Introduction

This Chapter details the implementation of the different modules discussed in the previous design section. The same plan will be approximately followed, adding nonetheless a section to define briefly the numerous technologies employed, composed of software, servers and languages. Then, the modules implementation will be described and detailed. It is composed of the construction of the main web site, and the creation of a web creation module and a web site manager. Finally, the use and design of the database will be described.

4.2 Technologies employed

When a dynamic web site has to be created, many tools and technologies are involved. By being dynamic, the implementation requires much more than just authoring tools. The most relevant tools were used each time it was needed to obtain the best efficiency during the implementation.

Here are the different tools used to develop the *easywebcreation* web site:

- **Microsoft Windows XP Professional:** The Operating System (OS), essential to run other technologies.
 - **EasyPHP (Apache + PHP + MySQL + PHPMyAdmin):** EasyPHP is a bundle designed to ease the installation of all the tools needed to run websites including server-side scripting and databases. This includes an Apache server for Windows, the PHP engine, MySQL engine and PHPMyAdmin, a tool to handle MySQL databases over the web that will be detailed later on.
 - **Macromedia DreamWeaver 4 UltraDev:** This is the web-authoring tool used to create the pages and templates. The UltraDev version provides the advantage of being capable to handle PHP. In fact, it was not a great functionality to use but allowed to see resultant web pages including PHP, what is not possible with DreamWeaver.
 - **Macromedia Extension Manager:** This tool allowed adding PHP module to DreamWeaver.
 - **PHAkt-PHP Server Model:** The module used to run PHP on DreamWeaver.
 - **Macromedia HomeSite 5:** This is a code-based editor that was used during the entire process. Actually, as soon as lot of PHP were included in the pages, it was not possible to use DreamWeaver any more.
 - **Adobe Photoshop 6 & Macromedia Fireworks 4:** The tools used to create the graphical design.
-

4.3 Modules Implementation

After the accomplishment of all the previous studies and investigations, a domain name was chosen very quickly because it was important to know on which name all the services will rely. Of course, short and explicit names were tried but the web is huge and many domain names are already reserved or at exorbitant prices. Keeping in mind that the user had to remember the name whatever happens, *easywebcreation* was chosen and registered on the internet in 48 hours. Once it was done, the first thing to do was to take care of the web site implementation.

4.3.1 General Web Site and login module

The web site implementation is the first thing to take into account during the implementation. Actually, as explained before, it is the grouping of all relevant information and links to other pages to form a welcome page. As such, in a first time, it was useful to provide links to test files, and also to test if the redirection of the domain name was pointing on something.

The structure was designed under DreamWeaver to obtain a pure HTML code that will then be included in PHP functions. By doing that, it eases the development of pages and their updates. This really makes the difference on huge pieces of code and gives the opportunity to always call the same code for different pages. For instance, 304 lines of code simply representing the header of each page are replaced by this command:

```
<?
    include ("designparts.php");
    general_header ();
?>
```

The first PHP function include all the content of the file *designparts.php* and then the function *general_header()* contained in it can be called, and fulfill its function of displaying the header in the new file.

This kind of simple call to functions containing bits of pages has been used for each visual parts of the site. As such, the implementation is much more structured and it even becomes possible to create a new page with 5 lines of code.

When this task was accomplished, and it was relatively repetitive and as such easy, the login/register was envisaged.

This is one of the most complicated part of the project. Actually, it was probably too ambitious first but the idea finally became real. The idea was to create a system allowing people to register by the intermediate of a form, and then add their data to the database, in the *user* table. But instead of just adding this information, other information randomly created was added as well to be reused later on without having to reuse the user login and password. Projects created by users can be important and a minimum of security will certainly engender an additional gain of trust from them. Thus, when user register, a special variable called *total* is created using a RSA Data Security encryption algorithm [16] called *md5*. This creates a 32 characters hexadecimal number hashing the concatenation of the user login and password. As such, it is still possible to decode this number at will. So the idea was pushed a little further and as well as the *total* variable was created and stored on the database, in the

user table, a corresponding hash was randomly created. This one is called *session* and is stored with the corresponding total number in another table called *session*. The idea is that a variable had to be passed from pages to pages and if it was a variable corresponding to a specific user, anybody having access to it could modify data freely. With this method, the session number is totally random and it the only variable we pass from a page to another. Furthermore, a new one is recreated at each new login so that the security is perfect. It was probably a little bit too much complicated for the more humble purpose of the project but stop the development of something that has already been begun is seen as a important loss of time rarely benefic.

4.3.2 Management Module

When the main body of the implementation was created, it was easier to work on additional modules. The fact that the registration was working gave a totally new framework to the project because it meant that session variable can be passed from a page to another and then that it was possible to behave in a way adapted to a specific user because identified.

By its structure, it was easier and wiser to begin with the management module. As stated during the design process, the menu in the site manager is placed on top of the page and is always visible. Namely it gives access to each pages of configuration wherever the users goes in the management system. In early development, even when buggy, the frame stayed on top and gave the opportunity to go back to a working page or to close the site manager and come back to the easywebcreation index page. On another hand, it will give users the opportunity to see the achievement of their modifications and updates.

We will now describe the main management options proposed by the site manager:

- Left menu links (page names)

This is one of the first options created. Actually, it is not the most complicated but it took time to manage information and take advantage on the session variable, which caused lots of concern. This category is basically a form with eight fields to fill in, corresponding to the eight pages that will appear on the final user site. The difficulty that was encountered here for the first time was the recovery of user information from the session number. Appendix 2 shows the structure of the file. Once it was finally done (first lines of the code), the framework of this category was to submit the page and enter a state (when the pointer *ok* equals 1) in the PHP code that adds information to the database when the page was submitted. Being in that state, a connection to the database followed by an update of its appropriate fields was performed.

An option allowing the creation of URL links corresponding to page names was implemented in early stages but after too problematic difficulties, it was decided to fix the pages' names to a norm. The index is called *index.php* and then other pages *page1.php* to *page8.php*. This is only the file names and will have no influence on pages names.

- Top menu
-

For a better user mental model, it was first thought that the same kind of feature described before could also apply to another menu, placed on another location of the website. It was not hard to implement but it appeared to be useless and confusing for the user. Then, it will be replaced by a dynamically created one that will create a link to the homepage and another one directly to the email of the user. This way, alternatively placed links called “Home” and “Contact” will be created. This will happen in another following process as this one will disappear.

- File Upload

Systems of upload have been investigated and developed because it was an important feature to provide to the user. A basic prototype had been developed and it was working. It gave the opportunity to connect to a ftp server with your specific information and then upload a file to it. Unfortunately, the hosts changed before we finished it and the new host does not accept ftp connection. This idea was abandoned while seeing how to find a solution.

- Site properties

The site properties section of the site management module is the last to use the kind of structure seen with the left menu edition page (viewable in appendix 2). So even if it does not enter the data in the same table of the database, here the table is *properties*, there is a form with field to fill in and data is added to the database when the form is submitted.

- Colours and cascading style sheets

Compared to the systems pre-quoted, this one is much more complicated. The first concern is the colours. It was not easy at all to define a way of providing the users the choice of selecting one of the 216 colours of the web palettes. To do so, a pop-up window is opened thanks to JavaScript and display 216 colour squares that are clickable and update automatically the right field in return, as shown on *Figure 8*.

Figure 8: Colours and CSS editing menu

In addition to colours elements, the choice was made possible to choose the font family and the different links decoration. This information will be added to the database automatically. The important thing to notice for all this section is that

database is read and information is retrieved each time the page is reloaded to display up to date information and keep the user aware of the real situation.

- Page content editing

This is definitely the most difficult part of the creation/management modules. The system created is inspired of the point of view provided in some online forum systems. What has been implemented is a system of markers that are added with JavaScript and parsed after the page is submitted. JavaScript functions allow prompting a message box where the user will enter the text that needs a specific formatting. When submitting this message box, the text entered will be surrounded by specific and understandable markers. For instance, when clicking on the usual **B** signifying bold, this prompt will appear:



Figure 9: Script Prompt for page content editing

For instance, if the text is *Hello World* the resultant text will be *[B]Hello World[/B]*. The same process will happen for italic, underline and centre. In addition to that, it is also possible to select a particular font, size or colour from three different lists. Finally, specific functions have been implemented to recognize automatically URL and email addresses. In the same time, another function is meant to recognize carriage return, empty spaces and line feed. These are all based on the PHP functions `str_replace` or `preg_replace`, whose aim is to parse a variable and find the relevant information to replace on it. This is a very complicated process, and it can even seem to be incomprehensible, like the following command to replace URL contained in the variable *\$text* by their HTML link equivalent:

```
$text = preg_replace("/((http(s?):\\V)|(www\\.))([\\w\\.]+)/i", "<a href=\"http$3://$4$5\" target=\"_blank\">$2$4$5</a>", $text);
```

The syntax is complex but the help provided by the official documentation, it can be understandable. The first part between quotes corresponds to the string to change, and the second one to the form the replaced string has to respect. For instance, without going too far, the \$2, \$3, \$4, \$5 variables match with their equivalent in the first part of the `preg_replace`. The numbering follows certain logic and as such each number corresponds to the string surrounded by the same number of parenthesis. Then \$4 corresponds to `www` and \$5 to the actual web address that we cannot see here. This process is by far one of the most interesting of the project because it allows doing very powerful modifications.

4.3.3 Creation Module

The creation module uses the same functionalities as those stated before. Actually, the code of the management module has been adapted for this new purpose but is more or less the same. In this case, the module is implemented as part of the entire website and each step is dependant of the previous one. As it was explained in the design section, this is a linear process to guide the user from the beginning to the end of the creation.

4.4 Database Implementation

Database implementation was made easy by the fact that its structure was defined in the design section. Furthermore, MySQL databases are so often associated with PHP that there are specific tools to handle them with simplicity and success. One of them is PHPMyAdmin (See Appendix 3 for a preview of the interface), a “de facto” standard to manipulate data on the MySQL database. The interface is simple and the gain of time procured is really impressive. Its functionalities are defined in the following table coming from the official website.

PHPMyAdmin is intended to handle the administration of MySQL over the web. Currently it can:

- create and drop databases
- create, copy, drop and alter tables
- delete, edit and add fields
- execute any SQL-statement, even batch-queries
- manage keys on fields
- load text files into tables
- create and read dumps of tables
- export data to CSV values
- administer multiple servers and single databases

Table 2: phpMyAdmin features

Using this powerful tool, the management and editing of the database becomes very easy. A special effort has to be made to respect the type of each field and to make sure that the id has its option *auto_increment* enabled or nothing will work.

4.5 Conclusion

After making a brief overview of the tools and technologies that was used, this chapter has detailed how the considerations stated in the design section have been implemented and how each steps of the implementation came to life. It has been particularly put in evidence that some difficulties have raised, especially due to free hosting facilities. On another hand, it was shown that reuse of pieces of code was lightening and then improving the entire development process. This trend has been used as much as possible in all categories of the development. Finally, even if the implementation showed particularly difficult parts to implement, conclusions have raised that it was sometimes not capable of performing all tasks, and then JavaScript functions have been implemented as well to cope with the weakness of PHP.

5. Evaluations

5.1 Introduction

This chapter will discuss how usability studies were conducted during the whole process and what benefits they allowed to obtain. As investigated at the beginning of the project, different studies will be deployed and results will be taken into account as much as possible. It will be a good and useful way to find out if the system meets the requirements and functions correctly. Because of the lack of time and the difficulties finding the exact audience required to submit them to a broad range of people; questionnaires were not performed. However, efforts were concentrated on heuristic evaluation, feature inspection and ethnographic studies to have reasonable conclusion about usability of the system.

5.2 Heuristic evaluation

The system has been submitted to three users who know about usability during the whole creation process. As we previously explained, these users are not typical users. They know about the project and have skills that will not particularly be found on users suiting the target audience defined in the primary investigations. However, it helped to find out at regular interval what the usability problems were. It was especially helpful in terms of website implementation. Users, on an observer point of view, had a better ability to determine where and when a usability issues was raised. Furthermore, the advantage of this technique is that it can be used at almost any time during the development cycle. By their expertise of the subject, users involved can even provide help at the beginning of the process. Then, based on explanations with storyboards, design specifications and paper mockups, the set of users selected was even capable to help and give advice before the real production work begins, which was of great help.

5.3 Feature inspection

Features inspection, in our case, was one of the most useful and rewarding usability testing. The website, created around modules and sub-modules, was often in an intermediate situation in terms of interface during the middle stages of the development, and finding a method to assess the usability of different components separately was of great worth. Then, to take advantage of this technique, little scenario were created to assess the right feature and see if the user realize exactly what is expected. That is how we discovered some issues in the managing system, especially with the top menu. Some links were not followed by the section number and as such, the linked pages were not behaving as concluded. With the same users, we discussed and raised the conclusion that the features allowing the modification of the website top menu was useless and can be deleted.

5.3 Ethnographical Studies/Field Observation

Ethnological studies are exactly the kind of studies needed at the end of a project. They give the opportunity to focus entirely on the user to have a last look at the usability issues. An evaluation like that is non-formal and it is then easier to see the situation as one that will happen in the real world, what is not necessarily feasible with traditional testing.

This is a great and rich experience, as it makes possible to see the user interacting with the system. On the other hand, users who want to be implicated in the process of evaluation have to be found. In this case, it was conducted on two persons. This is not a lot but it gave some results anyway and showed some weaknesses in the comprehension of the top menu in the site manager. This has been improved to meet user requirements.

5.4 Summary of studies

In this chapter, different methods have conducted to conclusive results in term of usability. Some of them have helped in decision making thanks to the help provided by users. That is essentially the case with features inspection that gave accurate user feelings about that they were testing.

Other studies gave different feedback, and generally that was especially big issues that were discovered because of the unfinished aspect of the project in development.

6. Conclusion

6.1 Evaluation of achievement

This section attempts to evaluate the project in terms of the aims and objectives that were identified from the beginning.

At the beginning, this project was specified to become a tool providing a different way of building web sites. Not one that will directly compete with professional state-of-the-art packages like Macromedia DreamWeaver, but most one that will be focused on helping people that don't find what they need in these packages. Actually, limitations were not really defined and the range of the project was decided by taste more than by deep thinking. The project had for main characteristics to be driven by server-side scripting languages (PHP) associated with a database (MySQL).

At first, the programming skills of the developer were not very good in PHP. To cope with that, many readings in tutorials and books have been carried out. The important thing was to begin with bases and to improve skills with time. Although the start was difficult because different languages, PHP and JavaScript, had to be learnt, progression became more regular across the weeks. Actually, if a change of host did not happened, changing a part of the system requirements, the website would probably have been finished a little bit earlier. This way, it would have given better opportunities to focus on deeper usability studies.

Anyway, the project has proved to be very interesting, and a great deal has been learned in the process of development. The website is working and provides powerful creation process and site manager. The problem is that by lack of time, not all the expected functionalities have been implemented. Only one template is available and it is sad because two other are designed, but not yet implemented in the site. Another point that misses the requirement is that help is not very well provided on the site. Basic information to help users browse are displayed but it could really be improve.

Whether it is because of over-optimism in the choice of the project or the solution chosen, or because of the issues encountered, the website does not entirely meet the initial requirements. In the same time, the aim was to create a webpage dynamically and simply, using template if possible, and all that is achieved.

6.2 Direction for future work

By its numerous characteristics and the interesting subject of the project, the directions for future work are endless.

First of all, what will be done in the near future is to complete the missing part and make of *easywebcreation* a complete service a web creation online. This includes adding images, managing FTP connections, allowing the site files to be received by

mail on a ZIP file and many other functions. With a little bit more time, this web service can become even more valuable for users.

Apart from missing functionalities, finishing the two missing templates integration to the site and create other when possible will add credibility to the site.

Finally, the design of the portal could be improved to a professional looking site.

Appendices

Appendix 1: MySQL database tables architecture

Field	Type	Attributes	Null	Default	Extra
id	tinyint(4)		No		auto_increment
user	varchar(30)		No		
total	varchar(50)		No		
bgcolor	varchar(7)		No	#FFFFFF	
darkcolor	varchar(7)		No	#6699CC	
brightcolor	varchar(7)		No	#DDEEFF	
txtcolor	varchar(7)		No	#000099	
linkcolor	varchar(7)		No	#FFFFFF	
linkhovercolor	varchar(7)		No	#FF9999	
linkactivecolor	varchar(7)		No	#FF0000	
font_family	varchar(50)		No	Arial	
txtdecoration	varchar(20)		No	none	
hoverdecoration	varchar(50)		No		
activedecoration	varchar(50)		No		

Table 3: colors_css MySQL table

Field	Type	Attributes	Null	Default	Extra
id	int(11)		No		auto_increment
user	varchar(15)		No		
total	varchar(255)		No		
linkname1	varchar(16)		No		
linkname2	varchar(16)		No		
linkname3	varchar(16)		No		
linkname4	varchar(16)		No		
linkname5	varchar(16)		No		
linkname6	varchar(16)		No		
linkname7	varchar(16)		No		
linkname8	varchar(16)		No		

Table 4: config MySQL table

Field	Type	Attributes	Null	Default	Extra
id	tinyint(4)		No		auto_increment
user	varchar(30)		No		
total	varchar(50)		No		
page0	blob		No		
page1	blob		No		
page2	blob		No		
page3	blob		No		
page4	blob		No		
page5	blob		No		
page6	blob		No		
page7	blob		No		
page8	blob		No		
contact	blob		No		

Table 5: pages_content MySQL table

Field	Type	Attributes	Null	Default	Extra
id	int(3)		No		auto_increment
user	varchar(255)		No		
total	varchar(255)		No		
name	varchar(255)		No		
slogan	varchar(255)		No		
email	varchar(50)		No		
footer	varchar(30)		No	© 2002 Easywebcreation	
ftphost	varchar(100)		No		
ftpport	int(4)		No	0	
ftpuser	varchar(30)		No		
ftppasswd	varchar(20)		No		
ftpdire	varchar(100)		No		

Table 6: properties MySQL table

Field	Type	Attributes	Null	Default	Extra
user	varchar(255)		No		
total	varchar(255)		No		
session	varchar(255)		No		
time	bigint(20)		No	0	
id	int(10)		No		auto_increment
check	bigint(20)		No	0	

Table 7: session MySQL table

Field	Type	Attributes	Null	Default	Extra
num	int(10)		No		auto_increment
user	char(100)		No		
passwd	char(100)		No		
total	char(100)		No		
date	date		No	0000-00-00	
firstname	char(20)		No		
lastname	char(30)		No		
email	char(30)		No		
webtype	int(5)		No	0	

Table 8: user MySQL table

Appendix 2 :

```

<?php
//leftbuttons.PHP
include("connection.php");

    MYSQL_CONNECT($server,$login,$pass) or die("Connection
failed");
    mysql_select_db("$base") or die("Connection failed");

    $queryfirst="select total from $table where
session='$session'";
    $resultfirst = @mysql_query($queryfirst);
    $total= @mysql_result($resultfirst,0,"total");
if($ok==1){
    echo"Your values have been updated";
    $query="update config set
linkname1='$linkname1',linkname2='$linkname2',linkname3='$linkname3',
linkname4='$linkname4' where total='$total'";
    $query2="update config set
linkname5='$linkname5',linkname6='$linkname6',linkname7='$linkname7',
linkname8='$linkname8' where total='$total'";
    $result = MYSQL_QUERY($query);
    $result2 = MYSQL_QUERY($query2);
    mysql_close();
    $ok=0;
}
else{echo"Give names to buttons on your left menu:";
    $recovervalues="select * from config where total='$total'";
    $resultvalues=@mysql_query($recovervalues) or die("problem with
recovervalues");

    $linkname1 = @mysql_result($resultvalues,0,"linkname1");
    $linkname2 = @mysql_result($resultvalues,0,"linkname2");
    $linkname3 = @mysql_result($resultvalues,0,"linkname3");
    $linkname4 = @mysql_result($resultvalues,0,"linkname4");
    $linkname5 = @mysql_result($resultvalues,0,"linkname5");
    $linkname6 = @mysql_result($resultvalues,0,"linkname6");
    $linkname7 = @mysql_result($resultvalues,0,"linkname7");
    $linkname8 = @mysql_result($resultvalues,0,"linkname8");
}
echo"
<html>
<head>
<title>Left Menu edition</title>
<meta http-equiv='Content-Type' content='text/html; charset=iso-8859-
1'>
</head>

<body bgcolor='#000000' text='#000000' style='
color:rgb(72,72,123);font-family : Arial, Helvetica, sans-
serif;font-weight : bold;'>";

echo"<br><br>
    <table width='90%' border='0' cellpadding='0' cellspacing='0'
align='center'>
    <tr>
        <td width='15' valign='top' bgcolor='#FFCC66'><img

```

```

src='../pics/gif/upcornerleft_black.gif' width='15' height='15'></td>
    <td valign='middle' bgcolor='#FFCC66' width='100%'
height='25' align='center'><img
src='../pics/gif/middletitles/leftbuttons.gif' alt='' width='243'
height='9' border='0'></td>
    <td width='15' valign='top' bgcolor='#FFCC66'><img
src='../pics/gif/upcornerright_black.gif' width='15' height='15'></td>
</tr>
<tr>
    <td width='15' valign='top' bgcolor='#FFFFFF'>&nbsp;</td>
    <td width='95%' align='center' bgcolor='#FFFFFF'>
        <FORM ENCTYPE='multipart/form-data' METHOD='POST'
action='$PHP_SELF?session=$session&ok=1'>
        <table border='0' align='center' bgcolor='#FFFFFF'
cellpadding='0' cellspacing='0' width='100%'>
            <tr><td>&nbsp;</td></tr>

                <tr> <td class='news'>          Link Name 1:</td>
                <td>
                    <input type='text' name='linkname1' value='$linkname1'
size='44' style='color:rgb(255,180,0); background-
color:rgb(72,72,123);'><br>
                </td>
            </tr>
<!------- IDENTICAL FOR LINK NAME 2 to 7 ----->
                <tr> <td class='news'>          Link Name 8:</td>
                <td>
                    <input type='text'
name='linkname8' value='$linkname8' size='44'
style='color:rgb(255,180,0); background-color:rgb(72,72,123);'>
                </td>
            </tr>
        <tr>
            <tr><td>&nbsp;</td></tr>

                <tr> <td width='344' colspan='2' align='right'>
                    <INPUT TYPE='SUBMIT'
VALUE='Submit' style='color:rgb(255,180,0);background-
color:rgb(72,72,123);'>
                    <input type='reset' name='Cancel '
value='Cancel' style='color:rgb(255,200,0);background-
color:rgb(72,72,123);'>
                </td>
            </tr> </table></FORM> </td>
            <td width='15' valign='top'
bgcolor='#FFFFFF'>&nbsp;</td>
        </tr>
        <tr> <td width='15' valign='bottom'
bgcolor='#FFFFFF'><img src='../pics/gif/downcornerleft_white.gif'
width='15' height='15'></td>
            <td valign='bottom'
bgcolor='#FFFFFF' width='100%' height='15'><img
src='../pics/gif/spacer.gif' height='15'></td>
            <td width='15' valign='bottom'
bgcolor='#FFFFFF'><img src='../pics/gif/downcornerright_white.gif'
width='15' height='15'></td>
        </tr>

</table>

</body>
</html>"; ?>

```

Appendix 3: phpMyAdmin Interface

Home

- easywebcreation
 - colors_css
 - config
 - pages_content
 - properties
 - session
 - user

Database easywebcreation

table test has been dropped.

SQL-query : [\[Edit\]](#)
 DROP TABLE `test`

Table	Action	Records	Size
colors_css	Browse Select Insert Properties Drop Empty	5	2.6 KB
config	Browse Select Insert Properties Drop Empty	5	2.4 KB
pages_content	Browse Select Insert Properties Drop Empty	3	2.4 KB
properties	Browse Select Insert Properties Drop Empty	5	2.4 KB
session	Browse Select Insert Properties Drop Empty	5	2.6 KB
user	Browse Select Insert Properties Drop Empty	5	5.3 KB
Sum		28	17.7 KB

- [Print view](#)
- Run SQL query/queries on database easywebcreation [\[Documentation\]](#) :
 Show this query here again
 Or Location of the textfile : [Browse...](#)
- [Query by Example](#)
- View dump (schema) of database

Structure only
 Structure and data
 Data only

 Add 'drop table'

References

- [1] Ted Schadler. March 2002. Web Services: The Next Technology Thunderstorm. URL: <http://www.forrester.com/ER/Research/Brief/Excerpt/0,1317,14741,00.html>
- [2] William W. Lee, Diana L. Owens. April 2000. *Multimedia-Based Instructional Design: Computer-Based Training, Web-Based Training, and Distance Learning*. Jossey-Bass. p. 3
- [3] Nielsen//NetRatings. March 2002. Global Internet Index Average Usage. URL: http://www.nielsen-netratings.com/hot_off_the_net_i.jsp
- [4] National Statistics. February 2002. Internet Access. URL: <http://www.statistics.gov.uk/pdfdir/intacc0402.pdf>
- [5] Jakob Nielsen. February 2001. Are users stupid? URL: <http://www.useit.com/alertbox/20010204.html>
- [6] Stephen H. Wildstrom. Business Week Online. February 8, 2002. DreamWeaver: Beyond Basic Web Pages. URL: http://www.businessweek.com/bwdaily/dnflash/feb2002/nf2002027_8961.htm
- [7] Jpowered. Advanced WebSite designs. URL: <http://www.jpowered.com/website/>
- [8] ZyWeb. Webware platform. URL: <http://www.zy.com>
- [9] Netcraft survey. January 2001. URL: <http://www.netcraft.com/Survey/index-200101.html>
- [10] Linux Mag. September 2000. URL: http://www.linux-mag.com/2001-03/mysql_01.html
- [11] PHPMyAdmin project. MySQL database manager. URL: <http://www.phpmyadmin.net/>
- [12] Dumas and Redish. 1994. *A practical guide to usability testing*. Intellect
- [13] Jakob Nielsen. Online Writings on Heuristic Evaluation. URL: <http://www.useit.com/papers/heuristic/>
- [14] Jurek Kirakowski. Human Factors Research Group. *Questionnaires in Usability Engineering*. URL: <http://www.ucc.ie/hfrg/resources/qfaq1.html>
- [15] Browser statistics. May 2002. URL: <http://www.upsdell.com/BrowserNews/stat.htm>
-

[16] RFC1321. April 1992. *The MD5 Message-Digest Algorithm*. URL:
<http://www.faqs.org/rfcs/rfc1321.html>

Bibliography

Multimedia-Based Instructional Design: Computer-Based Training, Web-Based Training, and Distance Learning by William W. Lee, Diana L. Owens

Secrets of Successful Web Sites: Project Management on the World Wide Web by David Siegel

Creating killer web sites: The art of third generation website design by David Siegel

User-Centered Web Development by Jonathan Lazar

Ratings Analysis : The Theory and Practice of Audience Research (Lea's Communication Series) by James G. Webster, Patricia Phalen, Lawrence W. Lichty

Web Pages That Suck: Learn Good Design by Looking at Bad Design by Vincent Flanders, Michael Willis

Effective Web Design: Master All the Essentials by Ann Navarro, Tabinda Kahn

MySQL (OTHER NEW RIDERS) by [Paul DuBois](#), [Michael Widenius](#)

Details: <http://www.amazon.com/exec/obidos/tg/stores/detail/-/books/0735709211/reader/24/102-5318862-1852156#reader-link>

Design Rationale : Concepts, Techniques, and Use (Computers, Cognition, and Work)

by [Thomas P. Moran](#) (Editor), [John M. Carroll](#) (Editor)

A Practical Guide to Usability Testing

by [Joseph S. Dumas](#), [Janice C. Redish](#) 1999

Usability Engineering

by [Jakob Nielsen](#) 1994

Building Effective Web Sites

by [Raymond Frost](#), [Judy Strauss](#) 2001

Effective Web Design : Master the Essentials (TM)

by [Ann Navarro](#), [Tabinda Khan](#) 1998

Planning and Designing Effective Websites

by [Sue A. Conger](#), [Richard O. Mason](#), [W. T. Mason](#) 1998