

A WEB ENABLED SYSTEM FOR MATRIMONIAL SERVICES

BY

**R. U. EKANAYAKE
(03/AS/019)**

This thesis is submitted in partial fulfillment of the requirements for the
Special degree of Bachelor of Science in Computer Science &
Technology.

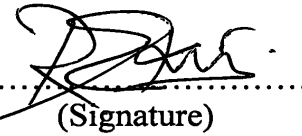
Department of Physical Science and Technology,
Faculty of Applied Sciences,
Sabaragamuwa University of Sri Lanka,
Belihuloya.

March 2009

DECLARATION

I declare that the work described in this thesis is, except where otherwise stated, entirely my own work carried out by me at the Department of Physical Science and Technology, Faculty of Applied Sciences, Sabaragamuwa University of Sri Lanka and Technology Exchange World (PVT) Ltd. under the supervision of Mr. R.A.K.R Kariemper and Mr. A.D. Janaka Pushpa Kumara. The thesis has not been submitted as an exercise for a degree at this or any other university.

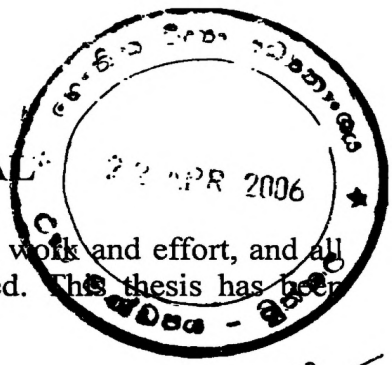
R.U. Ekanayake
(03/AS/019)



.....
(Signature)

22.01.2009
(Date)


CERTIFICATE OF APPROVAL



We hereby declare that this thesis is from the student's own work and effort, and all other sources of information used have been acknowledged. This thesis has been submitted with our approval.

Internal Supervisor:

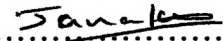
Mr. R.K.A.R. Kariapper
Lecture in computer science,
Faculty of Applied Sciences,
Sabaragamuwa University of Sri Lanka,
Belihuloya,
Sri Lanka.
Tel No: +94718080883.
E mail: kariapper.sab@gmail.com


.....
(Signature)

.....
20.04.2009
.....
(Date)

External Supervisor:

Mr. A.D. Janaka Pushpa Kumara
Project Coordinator,
Technology Exchange world (PVT) Ltd,
No: 128, High Level Road, Nugegoda,
Sri Lanka.
Tel No: +94-114560349/114556677.
E-mail: kumaraadjpk@gmail.com


.....
(Signature)

.....
2009/04/21
.....
(Date)



Head of the Department:

Dr. C.P.Udawatte
Head of the Department,
Department of Physical Sciences,
Faculty of Applied Sciences,
Sabaragamuwa University of Sri Lanka,
Belihuloya.
Tel No: +94-0773031087.
E-mail: chand@sab.ac.lk


.....
(Signature)

.....
2009.04.22
.....
(Date)

ACKNOWLEDGEMENT

I would like to thank many people for helping me during my Industrial training and it gives me great pleasure to convey my sincere gratitude to them all.

First, and foremost, I would like to deeply thank my external supervisor, Mr. Janaka Pushpakumara for his guidance, words of encouragement, kindness and patience. I shall never forget his support in the past months, without his supervision, there would not have been the steady progress in my web design development.

Also I am deeply indebted to my internal supervisor Mr. R.K.A.R Kariapper for invaluable guidance, encouragement, supervision and coordination given during the period of my research work and preparation of thesis.

Thanks also to the all staff members of TECHWORLD for providing opportunities to collaborate and learn. In particular, special thanks to chairmen of TECHWORLD, Mr. S. Jayawickrame for his assistance and advice.

Furthermore a special thanks to the academic staff of Faculty of Applied Sciences, SUSL for providing me to necessary academic insight.

I express my heart-felt gratitude to all my colleagues for them endless inspiration and heartiest assistance throughout the training period, and special notes to Mr.T.D.Rathnayake for sharing your knowledge and ideas.

I would also like to express my appreciation to all my wonderful friends who provided me love, caring, and inspiration during difficult times.

ABSTRACT

Wedding is an important as well as a memorable event of one's life. To make this moment more memorable one needs a lot of planning and organizing to make the wedding a cherished moment of life. One needs lots of assistance as well as expert advice to make everything perfect. The planning starts with hunt for a perfect life partner. Choosing a life partner is not an easy task. Earlier most of the marriages used to be arranged by family members, but the scenario is different these days. Current generation of youth want to make their own decisions, without being influenced by anyone. They always want it their own way. They believe in test kind of relationships, so not ready to choose a partner who is strange for him/her. Now the new trend is searching for life partner via the internet.

So, Technology Exchange World (PVT) Ltd planed to create an online matrimonial website as a solution for this. The web site is mainly focused to meet potential life partners via the internet and helps to do their wedding. So it is also act as an online wedding planer because it helps to arrange weddings, finding hotels, finding dresses, finding jewelers and cosmetics, as well as providing various services.

The Water fall model was used for the system development process. First feasibility study and then requirement analysis was carried out. Most of the requirements of the system that developed were captured in this phase by gathering requirements from Business Application experts in Technology Exchange World (PVT) Ltd. In the designing phase, user interfaces, database design and implementation of system was designed according to the requirements. The system was implemented using server side scripting language called PHP and the most popular open-source database system called MySQL. Coding was carried out using HTML, CSS, JavaScript and also AJAX technique was used for creating a better, faster, and more interactive web application. The components were tested individually and finally the integrated system was tested. Implementation proceeded module by module.

Result of the development process a dynamic and interactive online matrimonial website was found. The web site is providing great services to find life partners and arrange their weddings via the internet.

CONTENTS

DECLARATION.....	vii
CERTIFICATE OF APPROVAL.....	iv
ABSTRACT.....	v
ACKNOWLEDGEMENT.....	vi
CONTENTS.....	vii
LIST OF FIGURES.....	x
LIST OF TABLES.....	x

CHAPTER 1-INTRODUCTION

1.1 About the Technology Exchange World (PVT) Ltd, (TECHWORLD).....	1
1.2 Need for an online matrimonial website.....	1
1.3 About “ <i>tobemine.com</i> ” Matrimonial website.....	2
1.3.1 Users.....	2
1.3.2 Service Providers.....	3
1.3.3 Administrators.....	3
1.4 Objectives	3
1.4.1 Overall Objective.....	3
1.4.2 Specific objectives.....	3

CHAPTER 2-LITERATURE REVIEW

2.1 Software Engineering.....	4
2.1.2 Software Development Methods.....	4
2.1.3 Waterfall Modal	4
2.1.3.1 Requirement Analysis & Definition.....	5
2.1.3.2 System & Software Design.....	5
2.1.3.3 Implementation & Unit Testing.....	6
2.1.3.4 Integration & System Testing	6
2.1.3.5 Operations & Maintenance.....	6
2.1.4 What is rapid prototyping?	7
2.2 Applying Software Engineering Principles to a Web Site Development Process	7
2.2.1 Web Site Development Process	8

2.2.1.1 Requirements Definition/Specification	9
2.2.1.2 Requirements Analysis.....	9
2.2.1.3 Content Definition.....	9
2.2.1.4 Layout and Navigation Design.....	10
2.2.1.5 Human Computer Interface Design	10
2.2.1.6 Multimedia/Graphics Design	10
2.2.1.7 Implementation & Unit Testing.....	11
2.2.1.8 Integration & Testing.....	11
2.2.1.9 Operations & Maintenance.....	11
2.3 What is UML?	12
2.3.1 Goals of UML.....	13
2.3.2 Why Use UML?.....	13
2.3.3 Use Case Model	13
2.3.4 Class Diagram.....	14
2.4 Web Based Architecture.....	15
2.4.1 Basic Web Architecture	15
2.4.2 Web Architecture Extensibility.....	17
2.5 Web Database Applications: 3 tier Architecture.....	19
2.6 PHP (Hyper Text Preprocessor).....	21
2.6.1 Object Oriented PHP.....	22
2.7 What is a database?	23
2.7.1 Keys.....	24
2.7.2 Using PHP for database applications.....	25
2.7.3 Platforms, Servers, Databases, and PHP.....	25
2.7.4 Apache Web servers.....	25
2.7.5 Setting up own Web environment.....	26
2.8 MySQL.....	27
2.8.1 PHP with MySQL.....	28
2.8.2 Connecting to the database Using PHP and MYSQL.....	29
2.9 WampServer.....	29
2.9.1 Functionalities.....	30
2.10 Cascade Style Sheets (CSS).....	30
2.11 AJAX technology.....	31

CHAPTER 3-METHODOLOGY

3.1 Requirements analysis.....	32
3.1.1 Functional requirements.....	32
3.1.1.1 Users.....	32
3.1.1.2 Service providers.....	33
3.1.1.3 Administrators.....	33
3.1.2 Non-Functional requirements.....	34
3.1.2.1 Users.....	34
3.1.2.2 Service providers.....	34
3.1.2.3 Administrators.....	34
3.1.3 Use Case Diagram.....	34
3.1.4 Class Diagram.....	35
3.1.5 Detailed Specification.....	36
3.2 Design Modeling for the web application.....	39
3.2.1 Structure Design of the Web.....	40
3.2.2 Database Design.....	41
3.3 Implementation.....	41
3.3.1 Database implementation for the web.....	41
3.3.2 Web user interface implementation.....	44
3.3.2.1 Member Login Section.....	45
3.3.3 Coding and implementation.....	46
3.3.3.1 User Registration.....	46
3.3.3.2 Web Pages with AJAX technology.....	47
3.4 Testing.....	49
CHAPTER 4- DISCUSSION	52
CHAPTER 5- CONCLUSIONS AND RECOMMENDATIONS.....	54
REFERENCES.....	55

APPENDIX

Appendix A	Interface of the Home Page of “ <i>tobemine.com</i> ” website.....	56
Appendix B	Interface of the Registration Page of new user.....	56
Appendix C	User Profile	56
Appendix D	Service providing part of Home page.....	57
Appendix E	View of Members (Search Results).....	57
Appendix F	Administrator Panel	57
Appendix G	Messaging System for Users	58
Appendix H	Form for Add Wedding Proposals by Administrator.....	59
Appendix I	View of Wedding Proposals	59

LIST OF FIGURES

Figure 2.1	the Waterfall model	5
Figure 2.2	Iterative lifecycle of the Website-Waterfall model.....	12
Figure 2.3	a backend CGI program provides services to the WWW server on behalf of the client	18
Figure 2.4	Three Tier Architecture.....	20
Figure 2.5	trayicon of WampServer.....	30
Figure 3.1	the use case diagram for users in the <i>tobemine.com</i>	35
Figure 3.2	the Class diagram for users in the <i>tobemine.com</i>	35
Figure 3.3	Mapping user objectives in to interface actions.....	40
Figure 3.4	Member login section.....	45
Figure 3.5	Home Page of <i>tobemine.com</i>	46
Figure 3.6	Registration.....	46
Figure 3.7	View data of users by Administrator.....	48

LIST OF TABLES

Table 2.1	Advantages of PHP.....	22
Table 2.2	Advantages of Apache sever.....	26

CHAPTER 1

INTRODUCTION

1.1 About the Technology Exchange World (PVT) Ltd, (TECHWORLD)

TECHWORLD is a multi - disciplinary professional organization registered under the provisions of the Companies Act No. 7/2007 of Sri Lanka. TECHWORLD offers services, primarily in the fields of; Software and Security Systems Development, Web Designing and Development, Maintenance of Web Sites, Designing & Development of Computer based Games, Designing of Graphics; Logos, Banners, Visiting Cards, Letter Heads, Post Cards, Promotional and Ad. Material, Business Process Outsourcing (BPO), Project Management, Provision of Training in the fields of Information Technology and English Language Communication.

Its Vision is "*to achieve prosperity through developing Achievers*". They mobilize all resources at disposal to realize the vision of the company in an equitable and just manner. Policies of the organization are to deploy highest level of professionalism in achieving the objectives and goals of the company and to contribute towards sustainable development through application of Science and Technology.

TECHWORLD has been able to win the customer confidence through professional and disciplined deployment of its management skills, cost effectiveness and timely delivery of services and grow rapidly over a very short period.

1.2 Need for an Online Matrimonial Website

Wedding is an important as well as a memorable event of one's life. To make this moment more memorable one needs a lot of planning and organizing to make the wedding a cherished moment of life. One needs lots of assistance as well as expert advice to make everything perfect. The planning starts with hunt for a perfect life partner. Choosing a life partner is not an easy task. Earlier most of the marriages used to be arranged by family members, but the scenario is different these days. Current generation of youth is completely different. They want to make their own decisions, without being influenced by anyone. Whether it's matter of choosing their costumes, fashion accessories, cosmetics, girlfriends/boyfriends or life partner, they always want it their own way. So, choosing a life partner has become tough for current generation. They believe in tested kind of relationships, so not ready to choose a partner who is

strange for him/her. That is the reason behind the increasing popularity of matrimonial websites.

Online matrimonial websites provides comprehensive details of matching people. People can have glances of their positive attributes; they can even see the photographs listed in profile and makeup their mind in taking decision. These matrimonial websites provide you access of the contact details of people whom you are interested. The contacts are always secured, and no one can access without an approval from either you or matrimonial service provider.

Through online matrimonial websites people can get benefits of faster means of communication like emails, and chat which is a helpful factor in knowing each other. In brief, these matrimonial service providers are serving the needs of 21st century of people globally. The popularity of these kinds of matchmaking and dating websites are getting popular day by day due to their satisfactory and prompt services.

1.3 About “*tobemine.com*” Matrimonial Website

So, TECHWORLD planed to create an online matrimonial website that provides fabulous services. The web site is mainly focused to meet potential life partners via the internet and help to do their wedding. So it also act as an online wedding planer because it helps to arrange weddings, finding hotels, dresses, jewelry, cosmetics and provide various services.

The services provide by the website can be divided in to two parts:

- Help to finding potential life partners
- Help to finding services that used to arrange weddings

Three parties are involving on this website:

- Users
- Service providers
- Administrators

1.3.1 Users:

- Users can register as a member and create detailed profile.

That profile is the quickest way of letting others know about user. Just as user who more likely to express interest in a member with a detailed profile similarly others too would respond to her/his detailed profile. They also can add their photo.

1.3.2 Service Providers:

- Service providers can put their advertisements on the site.

Advertisements like Wedding shopping, Wedding hall & hotel, Studio service, Cake, Flower, Hair Beauty Salon and spa, Car Rental service, Wedding card design , Etc..... Those services like to include and relevant dealer will be registered by Admin and dealer.

1.3.3 Administrators:

- Administrator can handle all the functions in the web site.

Administrator can add, delete, maintain, update and view both user's profiles and service provider's profiles and also their accounts. As well as send messages to all users when necessary.

1.4 Objectives:

1.4.1 Overall Objective:

Implementation of the matrimonial web site is to providing a superior matchmaking experience by expanding the opportunities available to meet potential life partners through easy to use interfaces and features that can help them identify, filter and contact potential partners while protecting their privacy and security.

1.4.2 Specific objectives:

- Implementing a well designed up to date more secure web site.
- Literature survey about the software engineering.
- Literature survey about PHP, MYSQL.
- Literature survey about other latest web designing technologies.
- Designing and coding the system up to complete system including instance messaging, chat, and online payment gateways.
- Providing superior service to finding life partners users and arranging their weddings.

CHAPTER 2

LITERATURE REVIEW

2.1 Software Engineering

Software Engineering is more than just programming. Software Engineering is an engineering discipline which is concerned with all aspects of software production.

Software engineering is about creating high-quality software in a systematic, controlled, and efficient manner. Consequently, there are important emphases on analysis and evaluation, specification, design, and evolution of software. In addition, there are issues related to management and quality, to novelty and creativity, to standards, to individual skills, and to teamwork and professional practice that play a vital role in software engineering.

2.1.2 Software Development Methods

There are various software development approaches defined and designed which are used/employed during development process of software, these approaches are also referred as "Software Development Process Models". Each process model follows a particular life cycle in order to ensure success in process of software development. Following are the basic popular models used to this purpose.

- Waterfall model
- Rapid prototyping model
- Incremental model
- Synchronize-and-stabilize model
- Spiral model
- Object-oriented life-cycle models

2.1.3 Waterfall Model

The waterfall model is believed to have been the first process model which was introduced and widely followed in software engineering. The innovation was that the first time software engineering was divided into separate phases. In "The Waterfall" approach, the whole process of software development is divided into separate process

phases. The phases in Waterfall model are: Requirement Specifications phase, Software Design, Implementation and Testing & Maintenance.

All these phases are cascaded to each other so that second phase is started as and when defined set of goals are achieved for first phase and it is signed off, so the name "Waterfall Model". All the methods and processes undertaken in Waterfall Model are more visible.

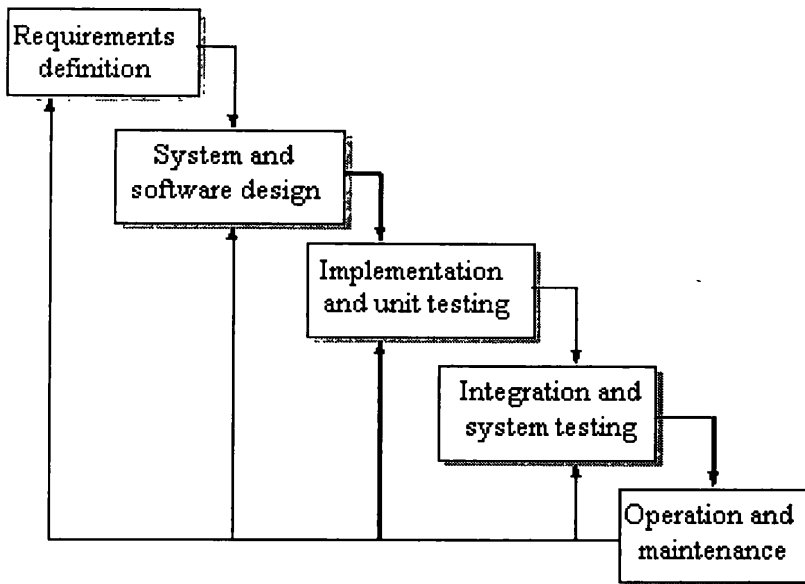


Figure 2.1 the Waterfall model

2.1.3.1 Requirement Analysis & Definition:

All possible requirements of the system to be developed are captured in this phase. Requirements are set of functionalities and constraints that the end-user (who will be using the system) expects from the system. The requirements are gathered from the end-user by consultation, these requirements are analyzed for their validity and the possibility of incorporating the requirements in the system to be development is also studied. Finally, a Requirement Specification document is created which serves the purpose of guideline for the next phase of the model. Requirements provide the foundation for the entire life cycle as well as for the software product.

2.1.3.2 System & Software Design:

Before a starting for actual coding, it is highly important to understand what we are going to create and what it should look like? The requirement specifications from first phase are studied in this phase and system design is prepared. System Design helps in specifying hardware and system requirements and also helps in defining overall

system architecture. This phase starts with the requirement document delivered by the requirement phase and maps the requirements into architecture. The architecture defines the components, their interfaces and behaviors. The deliverable design document is the architecture. The design document describes a plan to implement the requirements. The system design specifications serve as input for the next phase of the model.

2.1.3.3 Implementation & Unit Testing:

On receiving system design documents, the work is divided in modules/units and actual coding is started. Software implementation consists of implementing the requirements and design into code, data, and documentation. The system is first developed in small programs called units, which are integrated in the next phase. Each unit is developed and tested for its functionality; this is referred to as Unit Testing. Unit testing mainly verifies if the modules/units meet their specifications.

2.1.3.4 Integration & System Testing:

As specified above, the system is first divided in units which are developed and tested for their functionalities. These units are integrated into a complete system during Integration phase and tested to check if all modules/units coordinate between each other and the system as a whole behaves as per the specifications. In many software engineering methodologies, the testing phase is a separate phase which is performed by a different team after the implementation is completed. There is merit in this approach; it is hard to see one's own mistakes, and a fresh eye can discover obvious errors much faster than the person who has read and re-read the material many times. The system is delivered to the customer, after successfully testing.

2.1.3.5 Operations & Maintenance:

This phase of "The Waterfall Model" is virtually never ending phase (Very long). Generally, problems with the system developed (which are not found during the development life cycle) come up after its practical use starts, so the issues related to the system are solved after deployment of the system. Not all the problems come in picture directly but they arise time to time and needs to be solved; hence this process is referred as Maintenance.

2.1.4 What is rapid prototyping?

In many fields, there is great uncertainty as to whether a new design will actually do what is desired. New designs often have unexpected problems. A prototype is built to test the function of the new design before starting production of a product.

Building the full design is often expensive and can be time-consuming, especially when repeated several times building the full design, figuring out what the problems are and how to solve them, then building another full design. As an alternative, "rapid-prototyping" or "rapid application development" techniques are used for the initial prototypes, which implement part, but not all, of the complete design. This allows manufacturers to rapidly and inexpensively test the parts of the design that are most likely to have problems, solve those problems, and then build the full design.

In Software Engineering, a prototype generally refers either to a breadboard (or evolutionary) prototype or a throwaway (or one-off) prototype. Breadboard prototypes are often software in a development stage, focusing on a subset of the total requirements for a product. These prototypes usually are intended to evolve into the final design. Project managers may formally identify a software component as prototype to communicate with stakeholders that the component may or may not comprise the techniques ultimately allocated to the product design, or to meet business objectives. It should not be assumed that the prototype is merely for testing concepts (throwaway). That would be an aspect of a "research" project or "proof of concept." Prototypes provide the software developers with a "working model" for demonstration or use by customers, quality-assurance, business analysts, and managers to confirm or make changes to requirements, help define interfaces, develop collaborating components, and to provide proof of incremental achievement of scheduled contractual agreements. Software Prototyping serves any and all of these purposes in practice.

2.2 Applying Software Engineering Principles to a Web Site Development Process

Today, building effective web sites is not well understood, a trend that will continue for some time. There is not a formal development method being used in web development. Typically, the process for building web sites is "implement, test, and release". This is similar to the "code and test" process for software development. The problems of not having a web development process are obvious. Notice how many

sites are released with broken links, browser incompatibilities, and other problems. With growing complexity of web sites, a web site development process is needed. The goal of a web site development process is to formalize activities related to the development of web sites. The formalization of a development process aims at providing developers with a set of mechanisms which, when applied systematically, make it easier to obtain web sites of consistently high quality, in a predictable and reliable way.

The web development method comprises of a process that describes the steps to follow during development of web sites. The web site development process is driven by use cases, centered on the architecture, and deployed iteratively and incrementally using Unified Modeling Language (UML).

Use cases complement the object-oriented approach and form the basis of the web site development process. The use cases facilitate the implementation of web sites that satisfies the users' requirements. The web site architecture offers a global vision of the system. It describes the strategic choices that determine web site quality, such as reliability, adaptability, or the guarantee of performance, while making room for tactical decisions made during development.

2.2.1 Web Site Development Process

The process for developing web sites consist of applying software engineering principles to process will be called the Website-Waterfall model.

One software development approach that may be viable for some web site development is called Waterfall. Applying software engineering principles to a web site development process the Website-Waterfall model was created.

The Website-Waterfall model uses the iterative and incremental life cycle. In the iterative lifecycle, all iteration reproduces the Website-Waterfall lifecycle on a smaller scale. The objectives of iteration are established based on the evaluation of previous iterations. The iterative lifecycle revisits the various phases of the Website-Waterfall lifecycle several times.

This process describes the phases of web site development called Website-Waterfall model. All iteration comprises the following activities:

2.2.1.1 Requirements Definition/Specification

Requirements definition is the process by which the needs of the customer are translated into a clear, detailed specification of what the web site must do and achieve. The project specification must be in writing; a verbal agreement is not acceptable. That way, when the customer changes their mind, you can renegotiate the work. The output from this phase is the specification document. The specification defines what the project does, not how the project does it.

2.2.1.2 Requirements Analysis

The development team analyzes the requirements and specifications document for completeness and feasibility.

Once everyone agrees on the project specification, the next step is to analyze how to do it. The purpose of the analysis is to determine the scope of the effort. To help identify what is required for the project, it is necessary to answer to the following questions:

- What are the software tools necessary?
- What developer skills are required for the project?

A functional design document should be generated. This document consists of the information, figures and charts so that, when combined with the specification, someone new to the project can understand what the project does and how it does it.

2.2.1.3 Content Definition

Developing the content of the web site consists of identifying the audience, defining the need of the audience, and developing a statement of purpose. This purpose should then guide the development of the content on the web site.

2.2.1.4 Layout and Navigation Design

Laying out the navigation within the web site is the key. The user should be directed logically through the web site so that they always know where they are going. The layout and navigation design process categorize the information and creates a flowchart, which outlines both the organization of the web site, and the links, which will exist within the web site. The structure should naturally follow the content.

2.2.1.5 Human Computer Interface Design

Once it has been decided on the content, the organization of the web site has been outlined and navigation of the web site has been completed next the look of the web site should be addressed. The first step to designing the interface is to define what it is the user must accomplish. After defining the requirements, the next step is to determine the order a user would most likely use to accomplish the tasks. This requires understanding the target audience and how people want to do things. Once this is understood, the look and feel of the web site should be documented

An initial paper design must be generated. Depending on the complexity of the design an evaluation should be made on whether it is necessary to develop a mock-up system that users may beta test.

The documentation consists of the paper design of the major user interfaces and a flow diagram illustration how the interfaces interact with each other. If the user interface is simple enough, a text description of the interface is sufficient for documentation.

2.2.1.6 Multimedia/Graphics Design

The multimedia/graphics design involves representing the accessories that will be added to the web site. When deciding which accessories to add, be aware of how much time it's going to take to download. It must be determined if adding multimedia/graphics to the web site that the visual aspects of the web site will not distract users. The multimedia/graphics must be evaluated so that it is determined that the graphical content does not interfere with the readability of text.

A document listing the multimedia/graphics that will be used on the web site, as well as where it will be used should be generated.

2.2.1.7 Implementation & Unit Testing

During this stage, the process of building the web according to its design is done. The detail of the operations is implemented. The integration of the new code with existing code, issued from previous iterations, is implemented gradually during the construction. Unit testing procedures are applied to the prototype. Unit testing involves verifying that each unit meets its specification.

2.2.1.8 Integration & Testing

The individual program units or programs are integrated and tested as a complete system to ensure that the web site requirements have been met. After testing, the web site is delivered to the customer.

2.2.1.9 Operations & Maintenance

Normally this is the longest life cycle phase. Before the software gets deployed to the world, it must be documented properly. The on-line help is created and checked against the distribution web site. Someone needs to use the system aided only with the on-line help and if a user can operate the system just from the on-line help, then it is ready for deployment.

The web site is deployed and put into practical use. Maintenance involves correcting errors, which were not discovered in earlier stages of the life cycle; improving the implementation of the web site and enhancing the services as new requirements are discovered.

A visual view of the iterative lifecycle and how it revisits the various phases of the Website-Waterfall lifecycle several times is shown in figure 2.2.

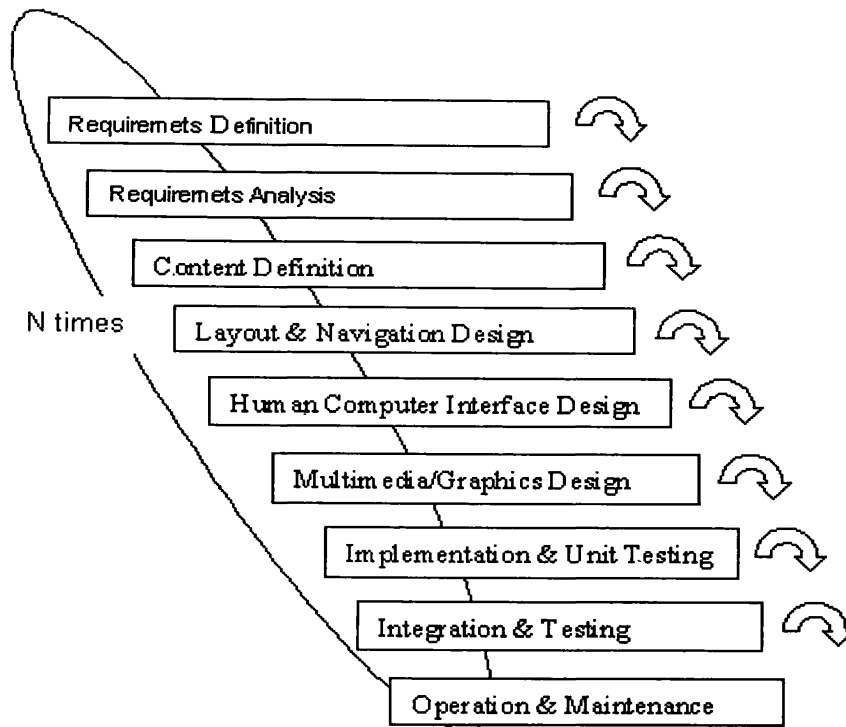


Figure 2.2: Iterative lifecycle of the Website-Waterfall model (Taylor C., 1999)

Using the website-waterfall model attempts to put flexibility into the waterfall model. The basic idea of the website-waterfall model is to encompass the best features of the classic waterfall model, while at the same time adding elements that is necessary for web development. (Carolyn Taylor,).

2.3 What is UML?

The Unified Modeling Language (UML) is a standard language for specifying, visualizing, constructing, and documenting the artifacts of software systems, as well as for business modeling and other non-software systems. The UML represents a collection of best engineering practices that have proven successful in the modeling of large and complex systems.

The UML is a very important part of developing Object Oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects. Using the UML helps project teams communicate, explore potential designs, and validate the architectural design of the software.

2.3.1 Goals of UML

The primary goals in the design of the UML were:

1. Provide users with a ready-to-use, expressive visual modeling language so they can develop and exchange meaningful models.
2. Provide extensibility and specialization mechanisms to extend the core concepts.
3. Be independent of particular programming languages and development processes.
4. Provide a formal basis for understanding the modeling language.
5. Encourage the growth of the OO tools market.
6. Support higher-level development concepts such as collaborations, frameworks, patterns and components.
7. Integrate best practices.

2.3.2 Why Use UML?

As the strategic value of software increases for many companies, the industry looks for techniques to automate the production of software and to improve quality and reduce cost and time-to-market. These techniques include component technology, visual programming, patterns and frameworks. Businesses also seek techniques to manage the complexity of systems as they increase in scope and scale. In particular, they recognize the need to solve recurring architectural problems, such as physical distribution, concurrency, replication, security, load balancing and fault tolerance. Additionally, the development for the World Wide Web, while making some things simpler, has exacerbated these architectural problems. The Unified Modeling Language (UML) was designed to respond to these needs.

Each UML diagram is designed to let developers and customers view a software system from a different perspective and in varying degrees of abstraction.

2.3.3 Use Case Model

The use-case model consists of actors and use-cases. Actors represent the customers or another software system. Use-cases represent the behavior of the system.

The use-case description shows how the system interacts step-by-step with the actors. The use-cases function as a unifying thread throughout the system's development cycle. A use case diagram shows us some of the use cases in system, some of the actors in system, and the relationships between us. A use case is a high-level piece of functionality that the system will provide. An actor is anyone or anything that interacts with the system being built. One of the major benefits of a Use Case diagrams is communication. Customers can look at this diagram and receive a great deal of information. By looking at the use cases, they will know what functionality will be included in the system. By looking at the actors, they will know exactly who will be interfacing with the system. By looking at the set of use cases and actors, they will know exactly what the scope of the project will be. This can help them identify up front any missing functionality.

2.3.4 Class Diagram

Class diagrams are widely used to describe the types of objects in a system and their relationships. Class diagrams model class structure and contents using design elements such as classes, packages and objects. Class diagrams describe three different perspectives when designing a system, conceptual, specification, and implementation. These perspectives become evident as the diagram is created and help solidify the design.

Classes are composed of three things: a name, attributes, and operations. Below is an example of a class. Class diagrams also display relationships such as containment, inheritance, associations and others.

The association relationship is the most common relationship in a class diagram. The association shows the relationship between instances of classes. For example, the class Order is associated with the class Customer. The multiplicity of the association denotes the number of objects that can participate in then relationship. For example, an Order object can be associated to only one customer, but a customer can be associated to many orders.

Another common relationship in class diagrams is a generalization. A generalization is used when two classes are similar, but have some differences. Class diagrams are

used in nearly all Object Oriented software designs. Use them to describe the Classes of the system and their relationships to each other.

Class diagrams are some of the most difficult UML diagrams to draw. To draw detailed and useful diagrams a person would have to study UML and Object Oriented principles for a long time. Before drawing a class diagram consider the three different perspectives of the system the diagram will present; conceptual, specification, and implementation. Try not to focus on one perspective and try to see how they all work together.

When designing classes consider what attributes and operations it will have. Then try to determine how instances of the classes will interact with each other. These are the very first steps of many in developing a class diagram. However, using just these basic techniques one can develop a complete view of the software system. (Gooch T., 2002).

2.4 Web Based Architecture

The World Wide Web (WWW) is one of the most popular and fastest growing phenomena in the history of computing. The Web is a global information-sharing architecture that integrates manifold online content and information servers in a fast, cost-effective, and easy-to-use-manner.

2.4.1 Basic Web Architecture

The basic web architecture is two-tiered and characterized by a web client that displays information content and a web server that transfers information to the client. This architecture depends on three key standards: HTML for encoding document content, URLs for naming remote information objects in a global namespace, and HTTP for staging the transfer.

HyperText Markup Language (HTML) - the common representation language for hypertext documents on the Web.

HTML is an application of the Standard Generalized Markup Language (SGML ISO-8879), an international standard approved in 1986, which specifies a formal meta-language for defining document markup systems.

An SGML Document Type Definition (DTD) specifies valid tag names and element attributes. HTML consists of embedded content separated by hierarchical case sensitive start and end tag names which may contain embedded element attributes in the start tag. These attributes may be required, optional, or empty. In addition, documents can be inter or intra linked by establishing source and target anchor points. Many HTML documents are the result of manual authoring or word processing HTML converters. HTML files are viewed using a WWW client browser (software), the primary user interface to the Web. HTML allows for embedding of images, sounds, video streams, form fields and simple text formatting. References, called hyperlinks, to other objects are embedded using URLs. When an object is selected by a hyperlink, the browser takes an action based on the URL's type, e.g., retrieve a file, connect to another Web site and display a HTML file stored there, or launch an application such as an E-mail or newsgroup reader.

Universal Resource Identifier (URI) - an IETF addressing protocol for objects in the WWW ("if it's out there, we can point at it"). There are two types of URIs, Universal Resource Names (URN) and the Universal Resource Locators (URL).

URLs are location dependent and contain four distinct parts: the protocol type, the machine name, the directory path and the file name. There are several kinds of URLs: file URLs, FTP URLs, Gopher URLs, News URLs, and HTTP URLs. URLs may be relative to a directory or offsets into a document. Arguments to CGI programs may be embedded in URLs after the "?" character.

HyperText Transfer Protocol (HTTP) - an application-level network protocol for the WWW. Tim Berners-Lee, father of the Web, describes it as a "generic stateless object-oriented protocol." *Stateless* means neither the client nor the server store information about the state of the other side of an ongoing connection. Statelessness is a scalability property but is not necessarily efficient since HTTP sets up a new connection for each request, which is not desirable for situations requiring sessions or transactions.

- In HTTP, commands (request *methods*) can be associated with particular types of network objects (files, documents, network services). Commands are provided for

- establishing a TCP/IP *connection* to a WWW server,
 - sending a *request* to the server (containing a method to be applied to a specific network object identified by the object's identifier, and the HTTP protocol version, followed by information encoded in a header style),
 - returning a *response* from the server to the client (consisting of three parts: a status line, a response header, and response data), and
 - *Closing* the connection.
- HTTP supports dynamic data representation through client-server negotiation. The requesting client specifies it can accept certain MIME *content types* and the server responds with one of these. All WWW clients can handle *text/plain* and *text/html*.

2.4.2 Web Architecture Extensibility

This basic web architecture is fast evolving to serve a wider variety of needs beyond static document access and browsing. The Common Gateway Interface (CGI) extends the architecture to three-tiers by adding a back-end server that provides services to the Web server on behalf of the Web client, permitting dynamic composition of web pages. Helpers/plugin-ins and Java/JavaScript provide other interesting Web architecture extensions.

- **Common Gateway Interface(CGI)** - CGI is a standard for interfacing external programs with Web servers (see Figure 2.3). The server hands client *requests* encoded in URLs to the appropriate registered CGI program, which executes and returns results encoded as MIME messages back to the server. CGI's openness avoids the need to extend HTTP. The most common CGI applications handle HTML <FORM> and <ISINDEX> commands.
 - CGI programs are executable programs that run on the Web server. They can be written in any scripting language (interpreted) or programming language (must be compiled first) available to be executed on a Web server, including C, C++, Fortran, PERL, TCL, Unix shells, Visual Basic, Apple script, and others. Security precautions typically require that CGI programs be run from a

specified directory (e.g, /cgi-bin) under control of the webmaster (Web system administrator), that is, they must be registered with the system.

- Arguments to CGI programs are transmitted from client to server via environment variables encoded in URLs. The CGI program typically returns HTML pages that it constructs on the fly.
- Some problems with CGI are:
 - The CGI interface requires the server to execute a program
 - The CGI interface does not provide a way to share data and communications resources so if a program must access an external resource, it must open and close that resource. It is difficult to construct transactional interactions using CGI.

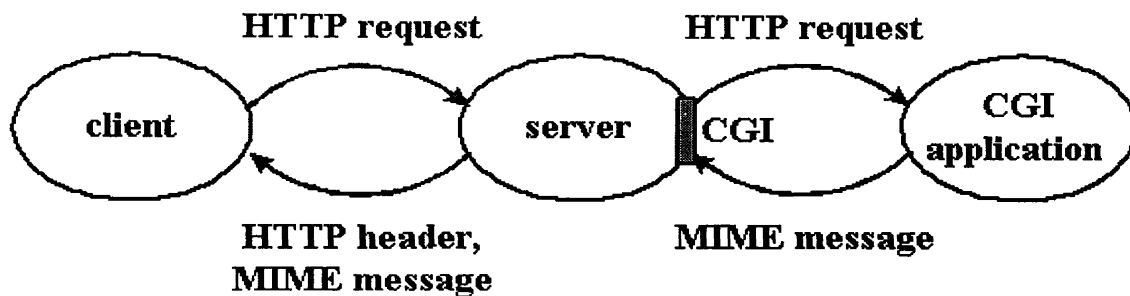


Figure 2.3 a backend CGI program provides services to the WWW server on behalf of the client (Hansen G. and Thompson C., 1997).

- **JavaScript** - JavaScript is a scripting language designed for creating dynamic, interactive Web applications that link together objects and resources on both clients and servers. A client JavaScript can recognize and respond to user events such as mouse clicks, form input, and page navigation, and query the state or alter the performance of an applet or plug-in. A server JavaScript script can exhibit behavior similar to common gateway interface (CGI) programs. JavaScript scripts are embedded in HTML documents using <SCRIPT> tags. Similar to Java applets, JavaScript scripts are directly interpreted within the client's browser and are therefore platform-independent.

2.5 Web Database Applications: 3 tier Architecture

Most applications deployed on the web implement a 3 tier architecture comprising of a database tier at the bottom, the application tier in the middle and the client tier on top.

The database tier consists of the DBMS or the database management system and the database. The complex application layer consists of the application logic and exchanges data between the different layers. The top layer is usually a thin client or a browser.

The Client tier which is usually a web browser processes and displays HTML resources, issues HTML requests and processes the responses. These web browsers are HTTP clients that interact with the Web servers using standard protocols. The client is often referred to as a thin client as very little application logic resides in the client tier. It does have built in features to do more than just display html pages, but they too are confined only to displaying data of various kinds. Some application logic in the form of JavaScript is executed by browsers, but its applications are limited in scope and nature. The advantage of a 3 tier model means that the thin client does not have to be configured and it does not have to depend on any operating platform or additional software. While there are subtle differences between the capabilities of different browsers, more or less they perform similarly.

The Middle Tier consists most of the application logic. It brings together the other layers of the 3 tier. It processes the inputs it receives from the clients and interacts with the database. This consists of the web server, web scripting language and the scripting language engine. While the Web server will process the HTTP requests and formulate responses, the calculation is done by the scripting language which runs on the scripting engine. The Web Server or the HTTP server listens for the HTTP requests on a network and return the responses that contain the requested resources. In cases where the request requires the output after running the program and interacting with the databases, the web server invokes the scripting engine to perform those tasks. The scripts that are executed by the scripting engine can be embedded into static HTML pages making client-tier integration easy. It allows for serving dynamic content and has built in libraries for fast access to the database tier.

The Bottom layer or the database layer consists of the Database Management System or the DBMS and the database itself. This database management layer manages the storage, retrieval of data as well as allows simultaneous access, provides security, data integrity and support applications. While a database is a collection of related data, a DBMS is a set of components for defining, constructing and manipulating the database. It consists of several components.

- a) Applications Interface or the libraries to communicate with the DBMS.
- b) SQL Interpreter which is a parser to check the syntax of incoming statements and translates them into the internal format.
- c) Query Evaluator that generates different plans for evaluating a query by considering database statistics and properties, selects one of these plans and translates it into low level actions that are then executed.
- d) Data Access: These are the modules that manage access to the data stored on disk, including a transactions manager, a recovery manager, the main memory buffer manager, data security manager and the file and access method manager.

A DBMS is required when there is a scenario which would involve multiple people accessing the data at the same time. While it may not be required where there isn't relational data or the data to be collected it atomic, it becomes a convenience when there are large amounts of data that require frequent updating. Together, these various tiers of a 3-tier architecture are the framework of most web applications deployed on the Internet.

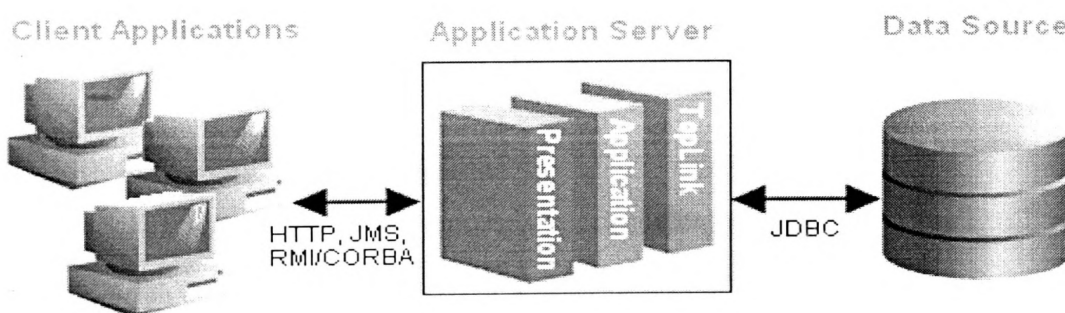


Figure 2.4 Three Tier Architecture

2.6 PHP (Hyper Text Preprocessor)

PHP, which stands for "PHP: Hypertext Preprocessor" is a widely-used Open Source general-purpose scripting language that is especially suited for Web development and can be embedded into HTML. Its syntax draws upon C, Java, and Perl, and is easy to learn. The main goal of the language is to allow web developers to write dynamically generated WebPages quickly. What distinguishes PHP from something like client-side JavaScript is that the code is executed on the server. If you were to have a script similar to the above on your server, the client would receive the results of running that script, with no way of determining what the underlying code may be.

PHP (Hyper text Preprocessor) is a language that has outgrown its name and is a widely used open source, general-purpose scripting language. It was originally designed for use in Web site development. In fact, PHP started life as Personal Home Page tools, developed by Rasmus Lerdorf to assist users with Web page tasks. It was originally conceived as a set of macros to help coders maintain personal home pages, and its name grew from its purpose. Since then, PHP's capabilities have been extended, taking it beyond a set of utilities to a full-featured programming language, capable of managing huge database-driven online environments.

PHP scripting code is frequently used to connect web pages to MySQL databases to create dynamic web sites. Some popular examples of PHP driven web sites would be blogs, message boards, and Wikis. PHP files are actually plain text files; they can be created in TextPad or any other Plain Text Editor. In a text editor set the Save As type to text and save the file with a .php extension. PHP files can also be created with Dreamweaver MX 2004. PHP is "open source" and therefore free. PHP is the most popular Apache module available, beating even ModPerl. It is a server-side scripting language usually written in an HTML context. Unlike an ordinary HTML page, a PHP script is not sent directly to a client by the server; instead, it is parsed by the PHP binary or module. HTML elements in the script are left alone, but PHP code is interpreted and executed and its code in a script can query databases, create images, read and write files, PHP proved so useful and popular, it rapidly grew to become the full-featured language popularity of PHP continues to grow rapidly because of its many advantages.

PHP is the most popular Apache module available, beating even ModPerl. It is a server-side scripting language usually written in an HTML context. Unlike an ordinary HTML page, a PHP script is not sent directly to a client by the server; instead, it is parsed by the PHP binary or module. HTML elements in the script are left alone, but PHP code is interpreted and executed and its code in a script can query databases, create images, read and write files, PHP proved so useful and popular, it rapidly grew to become the full-featured language popularity of PHP continues to grow rapidly because of its many advantages:

It's fast:	On Web sites, because it is embedded in HTML code, the time to process and load a Web page is short.
It's free	PHP is proof that free lunches do exist and that can get more than paid for.
It's easy to use	The syntax is simple and easy to understand and use, even for Programmers. For use in Web sites, PHP code is designed to be included easily in an HTML file.
It's versatile	PHP runs on a wide variety of operating systems — Windows, Linux, Mac OS, and most varieties of Unix.
It's secure	As long as scripts are designed correctly, the user does not see the PHP code.
It's customizable	The open source license allows programmers to modify the PHP software, adding or modifying features as needed to fit their own environments.

Table 2.1 Advantages of PHP

2.6.1 Object Oriented PHP

Object Oriented Programming in any language is the use of objects to represent functional parts of an application and real life entities. For example, a Person object hold the data related to a person and even provide some functionality that this person may be capable of. Object Oriented Programming has long been used in games to represent the objects such as a User or an Enemy, or even a Weapon. This amazing way of programming has proven just as useful in software and web development.

Any OOP language should have:

- Abstract data types and information hiding
- Inheritance
- Polymorphism

This can all be done using PHP classes:

```
<?php
class Something {
// In OOP classes are usually named starting with a cap letter.
var $x;

function setX($v) {
// Methods start in lowercase then use lowercase to separate
// words in the method name example getValueOfArea()
$this->x=$v;
}

function getX() {
return $this->x;
}
}
?>
```

2.7 What is a database?

A database is a collection of information or collection of interrelated and persistent data that is organized so that it can easily be accessed, managed, and updated. The most common type of database in use today is relational databases. In the most basic view, a relational database is simply a table of rows and columns, as shown in Table Each row as containing some "item and the columns in that row are parameters describing that item. In database terminology, each row is a record and each column is a field. A database column is defined to contain a specific type of data: strings, integers, dates, Booleans, and so on. The basic database operations include creating

tables, defining columns, and adding rows, deleting rows, changing values of a row, and searching for rows that match some condition.

Almost always, a database is accessed through a database server: a machine running the database software. Hence, to use a database, a connection must be established to such a server. After a connection is made, the database is sent commands, called queries, to which the database returns a result, usually in tabular form. Different database systems implement different methods of communicating with them

2.7.1 Keys

- A **simple key** contains a single attribute.
- A **composite key** is a key that contains more than one attribute.
- A **candidate key** is an attribute (or set of attributes) that uniquely identifies a row.
- A **primary key** is the candidate key which is selected as the principal unique identifier. Every relation must contain a primary key. The primary key is usually the key selected to identify a row when the database is physically implemented. For example, a part number is selected instead of a part description.
- A **super key** is any set of attributes that uniquely identifies a row. A super key differs from a candidate key in that it does not require the non redundancy property.
- A **foreign key** is an attribute or set of attributes that appears as a non key attribute in one relation and as a primary key attribute in another relation. Usually it is possible for a foreign key to also be the whole or part of a primary key.

2.7.2 Using PHP for database applications

PHP is particularly strong in its ability to interact with databases. PHP supports pretty much every database. PHP handles connecting to the database and communicating with it, so don't need to know the technical details for connecting to a database or for exchanging messages with it. When tell PHP the name of the database and where it is, and PHP handles the details. It connects to the database, passes instructions to the database, and returns the database response. Major databases currently supported by PHP such as dBASE, Informix, Ingres, Microsoft SQL Server, mSQL, MySQL, Oacle, PostgreSQL.

2.7.3 Platforms, Servers, Databases, and PHP

PHP is truly cross-platform. It runs on the Windows operating system, most versions of UNIX including Linux, and even the Macintosh. Support is provided for a range of Web servers including Apache (itself open source and cross-platform), Microsoft Internet Information Server, Website Pro, the iPlanet Web Server, and Microsoft's Personal Web Server. The latter is useful if want to test scripts offline on a Windows machine, although Apache can also be run on Windows.

PHP can be compiling as a standalone application. And can then call it from the command line. PHP is designed to integrate easily with databases. This feature is one of the factors that make the language such a good choice for building sophisticated Web applications. Many databases are directly supported, including Adabas D, InterBase, Solid, dBASE, mSQL, Sybase, Empress, MySQL, Velocis, FilePro, Oracle, UNIX dbm, Informix, and PostgreSQL. PHP also supports ODBC.

2.7.4 Apache Web servers

The software that delivers Web pages to the world is called a Web server. Several Web servers are available, but the most popular one is Apache. Approximately 60 percent of Web sites on the World Wide Web use Apache. Apache is open source software, which means it's free. It's available for all major operating systems. It's automatically installed with most Linux distributions and is preinstalled on Mac OS X. PHP is a project of the Apache Software Foundation, so PHP runs best with Apache.

Apache is generally best bet because it offers the following advantages:

It's free	Apache is open source software.
It runs on a wide variety of OS	Apache runs on Windows, Linux, Mac OS, FreeBSD, and most varieties of Unix.
It's popular	Approximately 60 percent of Web sites on the Internet use Apache, This wouldn't be true if it didn't work well. Also, this means that a large group of users can provide help.
It's reliable	After Apache is up and running, it should run as long as your Computer runs. Emergency problems with Apache are extremely rare.
It's customizable	The open source license allows programmers to modify the Apache software, adding or modifying modules as needed to fit their own environments. The open source license allows programmers to modify the PHP software, adding or modifying features as needed to fit their own environments.
It's secure	Free software is available that runs with Apache to make it into a secure SSL server. SSL is used to provide extra security for Web sites that need to protect important information.

Table 2.2 Advantages of Apache sever

2.7.5 Setting up own Web Environment

When starting a Web site from scratch, need to understand the Web site software fairly well and have to make several decisions regarding hardware and software. Also need to install a Web server and PHP, as well as maintain, administer, and update the system. Taking this route requires more work and more knowledge. The advantage is that have total control over the Web development environment.

The following are the general steps for setting up the Web environment

1. Set up the computer.
2. Install the Web server.
3. Install PHP.

2.8 MySQL

MySQL, the most popular Open Source SQL database management system, is developed, distributed, and supported by MySQL AB. MySQL AB is a commercial company, founded by the MySQL developers. It is a second generation Open Source company that unites Open Source values and methodology with a successful business model. It is written in C and C++.

- **MySQL is a database management system.**

A database may be anything from a simple shopping list to a picture gallery or the vast amounts of information in a corporate network. To add, access, and process data stored in a computer database, that need a database management system such as MySQL Server. Since computers are very good at handling large amounts of data, database management systems play a central role in computing, as standalone utilities, or as parts of other applications.

- **MySQL is a relational database management system.**

A relational database stores data in separate tables rather than putting all the data in one big storeroom. This adds speed and flexibility. The SQL (Structured Query Language) is the most common standardized language used to access databases.

- **MySQL software is Open Source.**

Open Source means that it is possible for anyone to use and modify the software. Anybody can download the MySQL software from the Internet and use it without paying anything. Anybody can study the source code and change it to suit their needs.

- **MySQL Server works in client/server or embedded systems.**

The MySQL Database Software is a client/server system that consists of a multi-threaded SQL server that supports different backends, several different client programs and libraries, administrative tools, and a wide range of application programming interfaces (APIs).

- **The MySQL Database Server is very fast, reliable, and easy to use.**

MySQL Server also has a practical set of features developed in close cooperation with users. The performance comparison of MySQL Server with other database managers on benchmark page MySQL Server was originally developed to handle large databases much faster than existing solutions and has been successfully used in highly demanding production environments for several years. Although under constant development, MySQL Server today offers a rich and useful set of functions. Its connectivity, speed, and security make MySQL Server highly suited for accessing databases on the Internet.

2.8.1 PHP with MySQL

PHP's MySQL and MySQL Improved functions are easy to use, if sometimes named confusingly. Each instance of a PHP/MySQL interaction must have a connection, a database select, and a query or command that returns a result identifier. The result identifier is like an ATM receipt that reports on the success or failure of an operation. There are several factors that make using PHP and MySQL together a natural choice:

PHP and MySQL work well together - PHP and MySQL have been developed with each other in mind, so they are easy to use together. The programming interfaces between them are logically paired up. Working together wasn't an afterthought when the developers created the PHP and MySQL interfaces.

PHP and MySQL have open source power - As they are both open source projects, PHP and MySQL can both be used for free. MySQL client libraries are no longer bundled with PHP. Advanced users have the ability to make changes to the source code, and therefore, change the way the language and programs work.

PHP and MySQL have community support - There are active communities on the Web in which developer can participate and they'll answer his questions. Developer can also purchase professional support for MySQL if he needs it.

PHP and MySQL are fast - Their simplicity and efficient design enables faster processing.

PHP and MySQL don't bog developer down with unnecessary details - developer doesn't need to know all of the low-level details of how the PHP language interfaces with the MySQL database, as there is a standard interface for calling MySQL procedures from PHP.

2.8.2 Connecting to the database Using PHP and MYSQL

The first step in a database interaction is connecting to the database. PHP function use to connect to the database. To make the connection, need function with four things:

Location: The database does not need to be on the same computer where PHP is installed. Therefore, need to tell the PHP connect function the name of the computer where the database is located (the hostname) either a domain name or an IP address. If the database is on the same computer as PHP can use localhost for the hostname.

User name: Must provide a valid user name that can be used to access the database. The database administrator sets this up. If using a Web hosting company, will be given a valid user name.

Password: Have to have a valid password to access the database. The database administrator sets this up. If using a Web hosting company, will be given a valid password for account.

Database name: An RDBMS can create and maintain many databases, so you need to tell it which databases you want to use. For security reasons, it's best to keep database connection information in a separate file and use it in PHP script with an include statement.

2.9 WampServer

WampServer is a Windows web development environment. It allows creating web applications with Apache, PHP and the MySQL database. It also comes with PHPMyAdmin and SQLiteManager to easily manage databases.

WampServer installs automatically (installer), and its usage is very intuitive.

WampServer is the only packaged solution that will allow reproduce production server. WampServer also has a trayicon to manage server and its settings.



Figure 2.5 trayicon of WampServer(www.wampserver.com)

2.9.1 Functionalities

WampServer's functionalities are very complete and easy to use.

- manage Apache and MySQL services
- switch online/offline (give access to everyone or only localhost)
- install and switch Apache, MySQL and PHP releases
- manage servers settings
- access logs
- access settings files
- create alias

2.10 Cascade Style Sheets (CSS)

Cascading Style Sheets (CSS) is a style sheet language used to describe the presentation (that is, the look and formatting) of a document written in a markup language. It's most common application is to style web pages written in HTML and XHTML, but the language can be applied to any kind of XML document, including SVG and XUL.

CSS is designed primarily to enable the separation of document content (written in HTML or a similar markup language) from document presentation, including elements such as the colors, fonts, and layout. This separation can improve content accessibility, provide more flexibility and control in the specification of presentation characteristics, enable multiple pages to share formatting, and reduce complexity and

repetition in the structural content (such as by allowing for table less web design). CSS can also allow the same markup page to be presented in different styles for different rendering methods, such as on-screen, in print, by voice (when read out by a speech-based browser or screen reader) and on Braille-based, tactile devices. While the author of a document typically links that document to a CSS style sheet, readers can use a different style sheet, perhaps one on their own computer, to override the one the author has specified.

CSS specifies a priority scheme to determine which style rules apply if more than one rule matches against a particular element. In this so-called *cascade*, priorities or *weights* are calculated and assigned to rules, so that the results are predictable.

The CSS specifications are maintained by the World Wide Web Consortium (W3C).

2.11 AJAX technology

Ajax, or **AJAX** (Asynchronous JavaScript and XML), is a group of interrelated web development techniques used to create interactive web applications or rich Internet applications. With Ajax, web applications can retrieve data from the server asynchronously in the background without interfering with the display and behavior of the existing page. The use of Ajax has led to an increase in interactive animation on web pages. Data is retrieved using the *XMLHttpRequest* object or through the use of Remote Scripting in browsers that do not support it. Despite the name, the use of JavaScript and XML is not actually required, nor do the requests need to be asynchronous. The acronym AJAX has thus changed to the term Ajax, which does not represent these specific technologies.

CHAPTER 3

METHODOLOGY

The system development process was carried out using the waterfall model. According to the stages of the Waterfall method, several steps were carried out. Those steps were ;

- Feasibility study and requirement analysis were done.
- The system was properly designed.
- Based on the system architecture which defines the main software blocks the software design was broken them further down into code modules.
- Based on the software design document the work was aimed to set up the defined modules and actual coding was started using PHP and MYSQL.
- Each unit was developed independently and was tested for its functionality. This is the so called Unit Testing.
- During integration the units which were developed and were tested for their functionalities. The modules were integrated into a complete system and were tested to check if all modules cooperate as expected.

3.1 Requirements analysis

Most of the requirements of system that developed were captured in this phase by gathering requirements from Business Application experts in Technology Exchange world (PVT) Ltd.

According to the requirements gathered, the system users can be divided in to 3 parts.

Those are:

- Users
- Service providers
- Administrators

3.1.1 Functional requirements

3.1.1.1 Users

- Users can register as a member of the system and create detailed profile.
- Users can search for a life partners and services provide by the system.
- User can visit website as a guest without registering with the site and can search for services provided by the system.

- User can also create expected partner profiles, interest expression on other members and accept interest that received from other users.
- Existing members can log on to the systems and add, delete and update their profiles.
- Users can visit services provide by the system and can view description of services also online booking and payment must be include.
- Users can communicate with system administrators when they have any doubts.
- Users can communicate with other members in the following ways.
 - Express interest in other members and accept or decline interest expressed by other members
 - Display user contact details and get contacted directly
 - Reply to messages received from other members
 - Chat with online members
 - Type user's own messages and send to other members
 - View validate contact details of other members
- Ranking of the users can be done according to the payments they pay and features provide to the premium members may be changed according to their ranking.

3.1.1.2 Service providers

- Service providers can put their advertisements on the site.
- Service providers can register as a member of the system and create detailed profile.
- Existing Service providers can log on to the systems and add, delete and update their profiles.
- Service providers can put their banners on the home page or other webpage of the site as their preference.
- Service providers can communicate with system administrators when they have any doubts.

3.1.1.3 Administrators

- Administrator can add, delete, update and view both user's profiles and service provider's profiles and also their accounts.

- Administrator can communicate with both users and service providers.
- Administrator can handle message system and send messages to both users and service providers.

3.1.2 Non-Functional requirements

3.1.2.1 Users

- Provide the site map for realize what are steps followed by user when registering, upgrade membership, searching and contacting with the web site.
- Only premium members can contact other members via chatting systems and sending emails. It is a special offer for the premium member.
- In searching results, premium members are the top listed members that displayed in result page.

3.1.2.2 Service providers

- Provide facility to add their own banners on the web site and according to the amount they pay, positioning of the banner is changed. Ex: If you pay lot, then banner will be put in home page.

3.1.2.3 Administrators

- Have responsibility to provide security and keep privacy data of members.
- Place wedding proposals of users that have less technical knowledge or have no much time to go through internet or shy people.

The categories of end users who will interact with the Web site were identified as part of the formulation and requirement gathering tasks.

The user categories (often called actors) provide an indication of the functionality to be provided by the web site and indicate a need for use- cases to be developed for each end-user (actor) were noted in the hierarchy.

3.1.3 Use Case Diagram

After identifying all the functional and non functional requirements, Content and functional requirements were listed and interaction scenario (use-case) written from the end-user's point of view were developed. The intent was to establish a basic

understanding of why the Web site was build, who will use it, and what problems it will solve for its users.

Rational Rose is an object-oriented Unified Modeling Language (UML) software design tool intended for visual modeling and component construction of enterprise-level software applications. So, Rational Rose was used to develop use-case diagram.

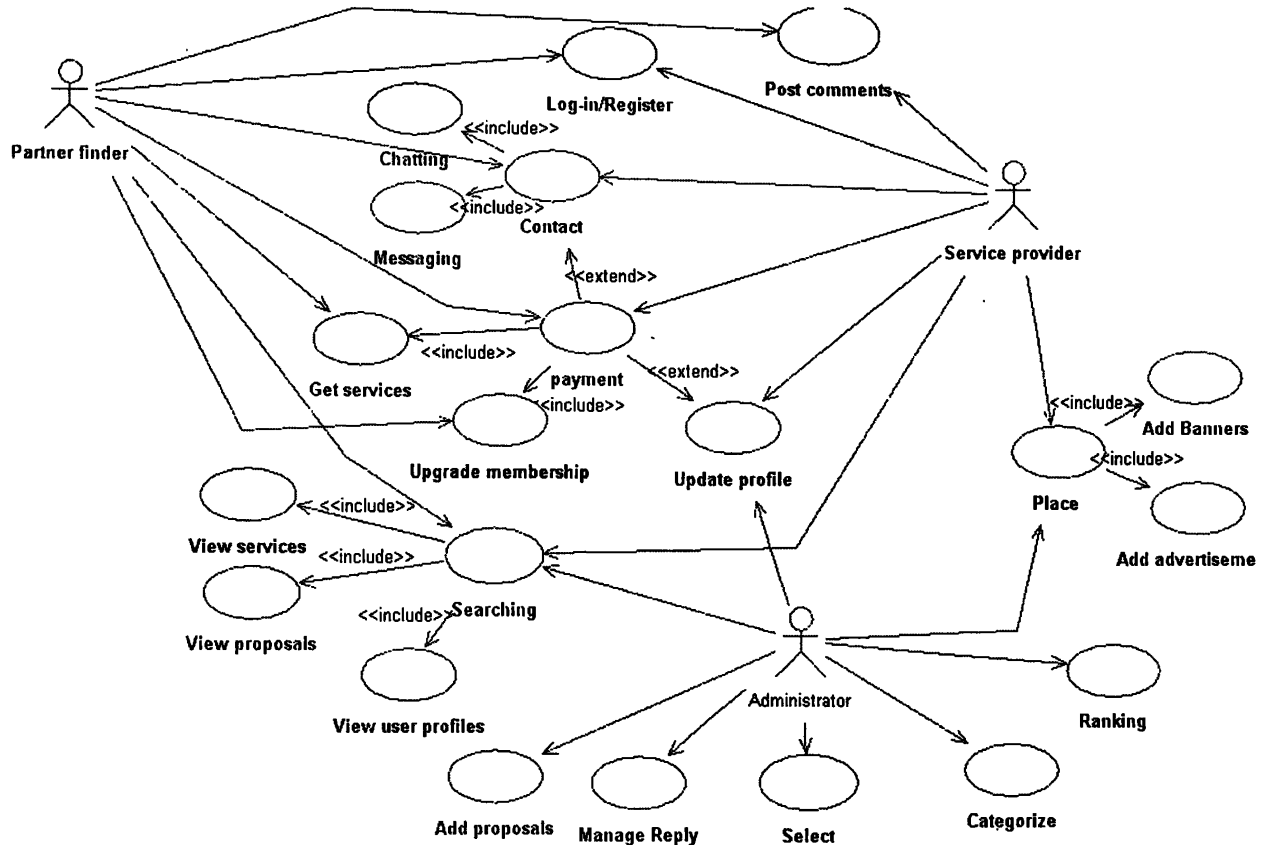


Figure 3.1 the use case diagram for users in the tobemine.com.

3.1.4 Class Diagram

The types of objects in a system and their relationships were identified using class diagram.

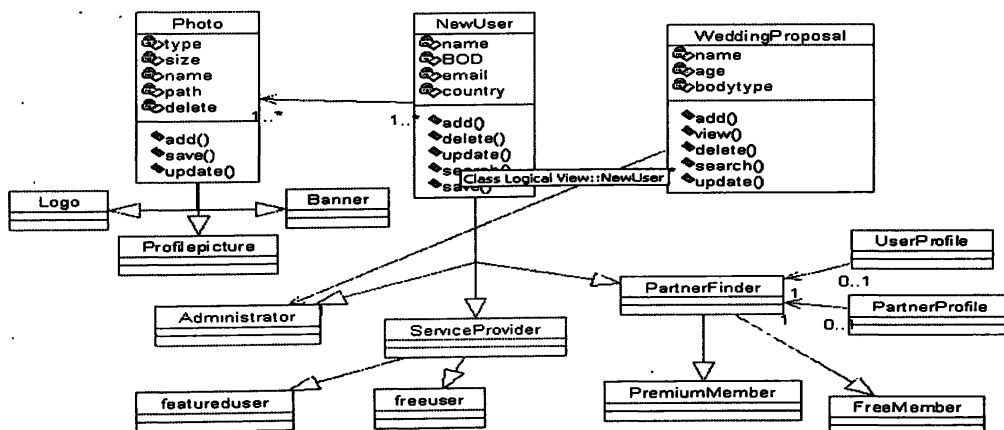


Figure 3.2 the Class diagram for users in the tobemine.com.

3.1.5 Detailed Specification

The last part of the requirements specification was a collection of design hints.

A). User can add profile and can maintain his own account

Registration Form's:-

If the user completely fills following fields then those data are wanted to login his profile. (All data are mandatory).

1. Profile id
2. email
3. confirm email
4. password
5. confirm password
6. contact number
7. gender
8. date of birth
9. religion./community
10. country of residence

Validation:

All data are mandatory.

Profile id – check whether it has been uniquely with other users.

Email – Must be in correct format.

User Profile:-

Basic information

11. Profile created by
12. gender
13. date of birth
14. marital status
15. have children
16. height
17. body type
18. complexion

19. special cases

20. blood group

Religious and basic background

21. religion

22. mother tongue

23. caste

24. sub caste/sect

25. gotra /gothram

26. manglik/kuja dosham

27. family values

Education and career

28. education

29. profession

30. annual income

Life style

31. diet

32. smoke

33. drink

Location

34. country of residence

35. state of residence

36. residency of status

Contact details

37. Contact number

38. Name of the contact person

39. Relationships with members

40. Convenient time to call

41. display options

Family

42. father

43. mother

44. brother

45. sister

More about yourself

- 46. Personality, long term goals, partner expectations and more
- 47.family background
- 48.free newsletters and alerts
- 49.current residence
- 50.city of residence

Then goes to thank you page where membership's payment option's shown and it is choice able.

Now starts for partner profile desired by user but optional (user can fill it later).

Mostly, same as user profile.

B). (User Login) when the user is logged to his profile:

- a. User will login and able to see the information added by him at the time of registration
- b. User will be able to edit his information with ability to add his photos also.
- c. User can send messages, emails to their interested ones and also can contact with administrators.

C). Advanced Search

Advanced Search will be on the bases of:

- 1. Quick Search (age, religion, country, language...etc)
- 2. City Search
- 3. Keyword Search
- 4. Smart Search

D). Success stories

Story given by clients with their photos.

E). Site map

Links of the pages with their little description.

F). Live Chatting

Chat application

G). Payment for membership

Online payment gateways or manual payment methods for premier members can be designed.

H). Admin panel

1. Ability to views the profiles of all the users and enable or disable them.
2. Ability to send newsletter to group or individual.
3. Ability to add advertisements of different sizes.
4. Content Management System
5. Multiple Levels of Admin Access may be included.
6. Discount Key Option.
7. Customer Support System (Email & Chat) (To be used in Call Centers).
8. Payment Integration.

I). Service Providers:

And also account handling (editing, updating) of service providers will be included on this system.

Divided in to three parts:

1. Reception site

From five star hotels to more economical reception halls must be included.

2. Shopping

Some services were categorized in to shopping list such as jewelry, Accessories... etc.

3. Services

Service providers: Beauticians, florists, caterers, photographers...etc.

3.2 Design Modeling for the web application

The first part of design was concerned with the architecture of the system. The system was broken in to sections that encompass broad group of functionality. A web application for matrimonial service was broken down in to a module that handles profile information, a module that handles services, and a module that handles

administrator's tasks. Also the Web site was broken down by the secondary pages. That is, the pages one click away from the home page. The "About Us" section serves to inform visitors about the company itself, while a site-map area is a resource for learning about the usage of Web site.

3.2.1 Structure Design of the Web

User interface design was begun with the identification of user, task, and environmental requirements.

- A rough sketch of the Web Application interface layout was developed.

An interface prototype (including the layout) was developed as part of the analysis modelling activity. A schematic first-cut layout of homepage sketch is shown in figure 3.3.

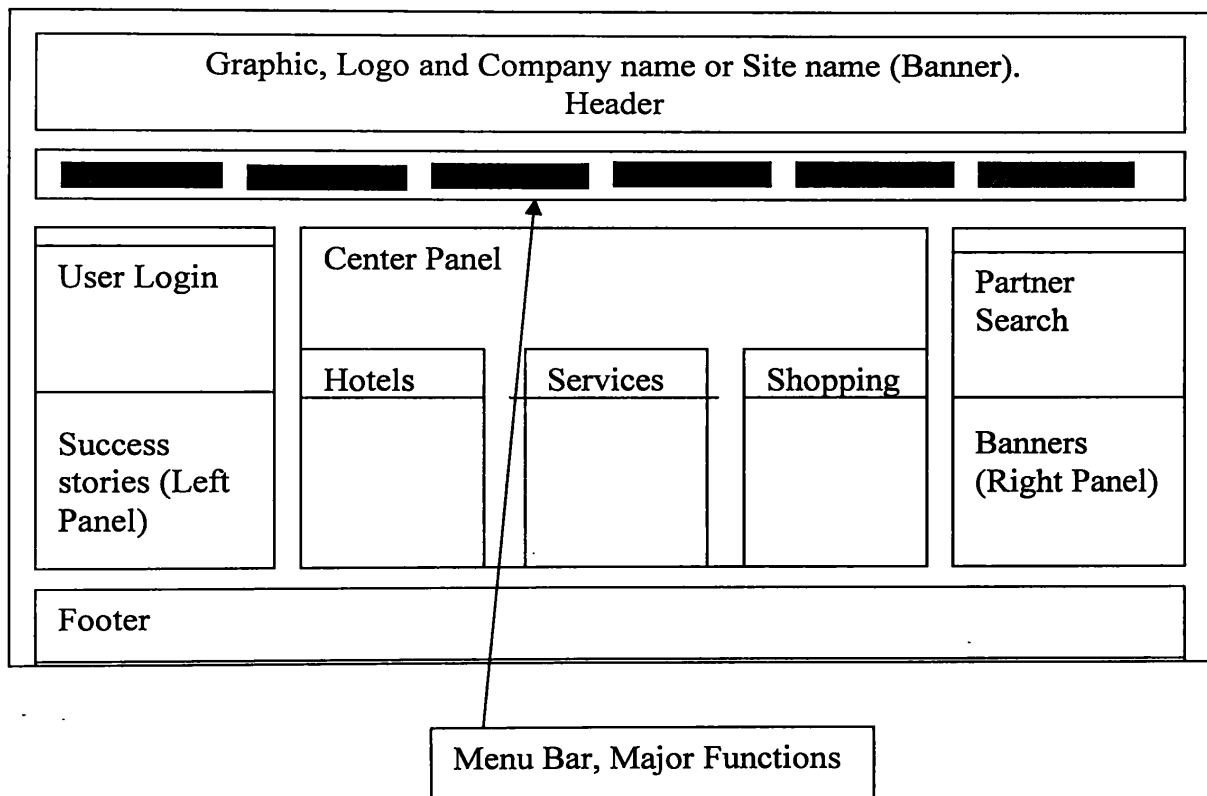


Figure 3.3 Mapping user objectives in to interface actions.

- User objectives were mapped in to specific interface actions.
- A set of user tasks that are associated with each action was defined.

Each interface action was associated with a set of user tasks. These tasks were identified during analysis modelling.

- Interface layout was refined using input of aesthetic design.

Content objects were identified (even if they have not yet been designed and developed)

- User interface objects were identified that are required to implement the interface.

3.2.2 Database Design

The process of doing database design generally consists of a number of steps. Not all of these steps will be necessary in all cases.

- Determine the data to be stored in the database
- Determine the relationships between the different data elements
- Superimpose a logical structure upon the data on the basis of these relationships.

Designing of database was done using class diagram and activity diagrams that created in requirement analysis part. On this stage Data Tables and relationship between those classes were identified.

3.3 Implementation

The system was built during implementation. Coding module was done in implementation phase. The implementation plan establishes the schedule and needed resources. It defines implementation details including programming languages, platforms, programming environments, debuggers, and many more. The implementation plan could be considered as part of the design, which is the position taken here or it, could be considered as the first accomplishment in the implementation phase.

3.3.1 Database implementation for the web

Database Wedding was developed in Wamp server and the used MySQL queries shown below. Part of the tables (admin_users and members) including in database was created using following queries.

```
CREATE DATABASE `wedding` DEFAULT CHARACTER SET latin1  
COLLATE latin1_swedish_ci;
```

```

USE `wedding`;
-----
--
-- Table structure for table `admin_users`
--
CREATE TABLE IF NOT EXISTS `admin_users` (
  `id` int(11) NOT NULL auto_increment,
  `user_name` varchar(20) NOT NULL,
  `password` varchar(10) NOT NULL,
  `state` int(11) NOT NULL default '0',
  PRIMARY KEY (`id`)
)
--
-- Dumping data for table `admin_users`
--
INSERT INTO `admin_users` (`id`, `user_name`, `password`, `state`) VALUES
(1, 'admin', '123', 1);
-----
--
-- Table structure for table `members`
--
CREATE TABLE IF NOT EXISTS `members` (
  `user_id` int(11) NOT NULL auto_increment,
  `profile_id` varchar(20) NOT NULL,
  `email` varchar(20) NOT NULL,
  `password` varchar(10) NOT NULL,
  .....
  PRIMARY KEY (`user_id`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1 AUTO_INCREMENT=2 ;

```

Also other tables were created in the phpmyadmin environment easily. Some tables were helped to increase dynamic features of the Website.

As an example: The country table includes all country names and their country codes. So, we can easily use that data with PHP codes when using that data on

<SELECT > (define a select list in HTML) using for loop. The coding was written as followed.

```

<SELECT>/*define a select list in HTML */
<?php
$rs = get_data("SELECT * FROM country ORDER BY country_name");
/* written query for select the country value*/
while($row=mysql_fetch_array($rs))
{
?>
    <option value="<?php echo $row['country_2_code']; ?>"><?php echo
$row['country_name'];?></option>/*****HTML        with        PHP
codes*****/
<?php
}
?> </SELECT>

```

So, we can use this function to get country values without repeating the long codes writing.

To connect to the database, several PHP functions were used. Those functions were modularized in to one file. If we want the database connection for a page we call this database file using INCLUDE function.

```

<?php function get_connection() //to get the connection
{
    $link = mysql_pconnect("localhost","root", "")or die("Could not connect: " .
mysql_error());
    mysql_select_db("wedding",$link) or die("Could not select database");

    return $link;
}
function if_exist($query)
    {
    }

/* ===== retriev data from a result set ===== */
function get_data($query)

```



```

        {
        }
/*=====saving method===== */
        function insert($query)
        {
        }
/*===== update===== */
        function update($query)
        {
        }
/*=====retriev data from a result set===== */
        function get_row_count($query)
        {
        }

```

?>

3.3.2 Web user interface implementation

The user interfaces were designed using HTML and CSS. An external style sheet was linked to an HTML document through HTML's **LINK** element:

```
<LINK REL=StyleSheet HREF="style.css" TYPE="text/css" MEDIA=screen>
```

```
/* CSS Document */
```

```

body {
    font: 12px Tahoma,sans-serif;
    color: #134b95; text-align: center;
    background-color:#E1E1FF; }
.spacer{
    font-size:0;
    line-height:0;
    clear:both;
}
#logo
{
    width:267px;
    height:100px;

```

```

        background:url(../../images/) 0 0 no-repeat;
        float:left;
    }

    #main_menu
    {
    }

#center_panel
    {
    }

```

3.3.2.1 Member Login Section

Member Login

Profile ID

Password



[Forget your password?](#)

Fig 3.4 Member login section

The following codes represent the interface design of the member login page. When programming a page with an HTML form, data are passed using POST method.

```

<form name="login" action="checking_member.php" method="post" >
    <label>Profile ID</label>
    <input type="text" name="profile_id" id="profile_id"/>
    <br class="spacer" />
    <label>Password</label>
    <input type="password" name="password" id="password"/>
    <br class="spacer" />
    <input type="image" src="images/go_btn.gif" title="Go" class="goBtn" />
    <br class="spacer" />
    <label class="large"><a href="#">Forget your password?</a></label>
    <br class="spacer" />
</form>

```

Using CSS and HTML languages, user interfaces were implemented dynamically and Photoshop, Dream viewer Software tools were used to create interactive Web site.



Fig 3.5 Home Page of tobemine.com

3.3.3 Coding and implementation

3.3.3.1 User Registration

Account Information
(You will need the following information to login later)

Profile ID:

Email:

Password:

Confirm Password:

Profile Information
(Enter information of the person looking for a life partner)

Gender: Male Female

Date of Birth: Day Month 1999
(visible only to you)

Religion / Community:

Country of Residence:

I agree to the Pathikula.com Privacy Policy and Terms and Conditions.

Register

Fig 3.6 Registration

An interface that implemented to registering new user is shown in a figure 3.7. Using POST method the data filled by the user send to the PHP file that returning those post values to the function of an object created by the user class. The coding that used to initialize the user class is given below.

```

class User{
    var $profile_id;
    var $email;
    var $password;
    .....
    function User($profile_id)
    {
        $rs = get_data("SELECT * FROM members WHERE
profile_id = '$profile_id'");
        while($row=mysql_fetch_array($rs))
        {
            $this->profile_id=$row['profile_id'];
            $this->email=$row['email'];
            $this->password=$row['password'];
            .....
        }
    }// end constructor

} // end user

```

In this class, check whether the password, email format and profile id are in the correct form by using several functions.

3.3.3.2 Web Pages with AJAX technology

With AJAX, JavaScript can communicate directly with the server, using the JavaScript **XMLHttpRequest** object. With this object, JavaScript can trade data with a web server, without reloading the page.

AJAX uses asynchronous data transfer (HTTP requests) between the browser and the web server, allowing web pages to request small bits of information from the server instead of whole pages. The AJAX technique makes Internet applications smaller, faster and more user-friendly.

Profile ID	plum456
Create By	Parent / G
Gender	Female
Marital Status	Divorced
Have Children	
Body Type	Athletic
Complexion	Wheatish
Height	4ft 6inc - 137cms
Mother Tongue	English
Caste	Hindu:Brahmin - Sara
Manglik	No
Special Cases	None
Personal Values	Liberal
Diet	Veg
Smoke	No
Drink	No
Religion	
Community	Sanskrit
Education	Less than high schoo

User_ID	Profile_ID	E-mail	Last Login Date
12	rathu99	rathu@gmail.com	2009-01-07
13	rati67	wer@yahoo.com	2009-01-07
14	rathu991	rat9@das.com	2009-01-07
15	ratu1	man6@gh.com	2009-01-07
16	upumalee	upuli@gmail.com	2009-01-07
17	we	er@ty.io	2009-01-08
18	rath6	tar5@ja.bom	2009-01-08
19	rt89	tar5@ja.bom	2009-01-08
20	56ui	tar5@ja.bom	2009-01-08
21	aruni	aruna@ga.won	2009-01-08
22	asd	as@gh.com	2009-01-08
23	wer	wer@f.bmm	2009-01-09
24	upli456	upli@my.com	2009-01-31
25	pani890	pan567@ham.com	2009-01-31
26	rathu1239	tar@hon.com	2009-02-02
27	uoi123	rath@ulo.nom	2009-02-02
28	tharindu	tar@yahoo.com	2009-02-02
29	plumi	p@hon.lon	2009-02-02
30	janu	janu23@fon.com	2009-02-03
31	thush	tush@k.lv	2009-02-03
32	ratul	rat@gh.kk	2009-02-03
33	rathu123	tar@gh.kk	2009-02-03
34	plum456	plum@gmail.com	2009-02-04
35	chaturangi	chaturangi@com	2009-02-26

Fig 3.7 View data of users by Administrator

In the fig 3.8 shown that, if administrator clicks on one row in left table, the result shows according to that user in the right table on the same page.

AJAX coding according to that task is given below.

```

var xmlhttp;
function showUser(str)
{
}
function stateChanged()
{
if (xmlhttp.readyState==4 || xmlhttp.readyState=="complete")
{
}
}

function GetXmlHttpRequestObject()
{
var xmlhttp=null;

```

```

try
{
// Firefox, Opera 8.0+, Safari
xmlHttp=new XMLHttpRequest();
}
catch (e)
{
//Internet Explorer
try
{
xmlHttp=new ActiveXObject("Msxml2.XMLHTTP");
}
catch (e)
{
xmlHttp=new ActiveXObject("Microsoft.XMLHTTP");
}
}
return xmlHttp;
} // JavaScript Document

```

3.4 Testing

Testing is the practical, hands-on process of checking that a web site meets the design specifications (in terms of information, interactive and visual design) and usability criteria (finding information easily, accessibility and compatibility). It is a vital and essential part of the web development process.

Most web sites are not tested adequately. As a consequence, accessing the site can be a minefield of problems for users, such as:

- slow response and download times
- missing content, broken images or error messages
- patchy or unstable performance in some platforms and browser software versions

- information that is inaccessible to people using adaptive technologies such as screen readers and screen magnifiers
- visual layouts that break or are inconsistent, and text that is illegible due to sizing problems

The Web site was tested to make sure that users get what we intended from the site. The Web site usually provides information, promote a product or service or attract new clients. Site testing makes sure that the site was designed so that users can find information and make contact with administrator easily. It reflects badly if information is difficult to find on the site, and the user has a bad experience. Users often pass their impressions on to other people, particularly if those impressions are negative ones.

Testing plan was directed the testing process and was ensured that:

1. the developing web site meets the client's original requirements
2. all usability issues are addressed
3. layout and consistency issues are fixed
4. problems are identified and resolved early

Larger web sites with a more complicated structure or interactive components often undergo technical development first. One of the first stages of site testing was functionality and usability testing.

Usability testing at this stage was extended to testing the on focus groups from the site's target audience. Feedback was collected on how appropriate and easy to use the site was. Then we used these results to improve the functionality of the site and to inform further content development and visual and interactivity design.

Code (HTML, javascript, ColdFusion, CSS, etc.) is what makes the web site function. Technical testing takes place as the site's code is being developed. It includes:

- testing the functionality of the code

- ensuring that the code meets current accepted web standards
- checking the code's compatibility with different browser versions (see Compatibility testing)

Compatibility testing takes place during technical and creative development. It includes checking that the visual and interactive elements, as well as other functionality (eg. search functions, video streaming, animation, etc.), are compatible with a range of platforms and browser software versions.

The same web site can behave quite differently in different platforms and browsers. The behavior of the site was checked in a range of currently used browser versions. The testing process was used to accommodate these differences into the web design.

As the site reaches completion, the html and other code will be checked, the content, information architecture, visual design and functionality of a web site overall to make sure:

- web page content is complete, correct and proofread
- hypertext and navigational links work and link to the correct Internet locations
- site navigation is logical, efficient and easy to use
- basic accessibility guidelines have been followed
- graphic, visual and interactive elements and other functionality are compatible with a range of platforms and browser software versions
- the site works efficiently and easily as a whole

Once any remaining bugs have been identified and resolved, the web site is ready to send live to the Internet.

CHAPTER 4

DISCUSSION

During my project period, I act several roles as a designer, as a developer and as a tester to achieve the aim of this project. The aim of this project was to create a user friendly, attractive and profitable online matrimonial site that helps to find partners and arrange their wedding.

I was responsible for implementation of overall system without chatting system. My first stage of this development process was to choosing a suitable web development process. Then I used Website-waterfall model and prototype model.

The WebSite-Waterfall model uses the iterative and incremental life cycle. In the iterative lifecycle, all iteration reproduces the WebSite-Waterfall lifecycle on a smaller scale. The objectives of iteration are established based on the evaluation of previous iterations. The iterative lifecycle revisits the various phases of the WebSite-Waterfall lifecycle several times.

After building the specification, work on the web site was scheduled upon receipt of the signed proposal, a deposit, and any written content materials and graphics we wish to include. Here normally the layouts and navigation was designed as a prototype.

Some customers may be interested only in a full functional prototype. In this case we may need to show them the interactivity of the application or site. But in most of the cases customer may be interested in viewing two or three design with all images and navigation.

There can be a lot of suggestions and changes from the customer side, and all the changes should be freezed before moving into the next phase. The revisions could be redisplayed via the web for the customer to view.

To enhance the security of the system, I used some techniques. Those are,

- User authentication was done in PHP using session management feature.
- The password of the user was encrypted using PHP function called md5 ().

As a suggestion we can destroy the session after a specified time. Then after session is timeout, user has to re-login to the system.

It is helpful to make use of object oriented programming (OOP) in an application where the data are largely hierarchical.

In many ways the include function is like a function call. Given the name of a file, PHP attempts to parse the file as if it appeared in place of the call to INCLUDE. The difference from a function is that the code will be parsed only if the INCLUDE statement is executed.

As library of function grows, we may discover some inter dependencies. Imagine a module for establishing a connection to a database, plus a couple of other modules that rely on the database connection. Each of these two scripts will include the database connection module. But what happens when both are themselves included in a script? The database module is included twice. These may cause a second connection to be made to the database, and if any functions are defined, PHP will report the error of a duplicate function.

The same web site can behave quite differently in different platforms and browsers. The behavior of the site was checked in a range of currently used browser versions. (Microsoft Internet Explorer and Mozilla Firefox) The testing process was used to accommodate these differences into the web design.

Once any remaining bugs have been identified and resolved, the web site is ready to launch in the Internet. Maintaining part is essential to make the Website more user-friendly and also we must keep the software up to date with changes in technology.

CHAPTER 5

CONCLUSIONS AND RECOMMENDATIONS

The goal of the online matrimonial web site has been achieved and I did my best to release the web site with out broken links, browser incompatibilities, and other problems.

Finally the “*tobemine.com*” web site was found and it can be concluded that web site is most functional, usable, reliable, safe, security, maintainable, portable, efficient, accuracy, and lack of error proven solution.

The web site is providing great services to search life partners, express interest if someone interesting on some other one, contact partners and arrange their weddings via the internet.

Continuous monitoring and testing are highly recommended in order to check whether the system meets its goals in the long run.

The following can be suggested for the further enhanced of the system

1. Implement the online video/text chatting system to both administrators and users.
2. Implement the online payment gateway system to carry out all payments via the internet.
3. Enhance the security of the web site, as an example; highly restrict the system that user to copy the images by right clicking.
4. RSS news feed system willing to be added to the site to view latest news.

REFERENCES

- Atkinson L. (2001), *Core PHP programming*, 2nd Edition, Addison wesley longman (Singapore) Pvt. Ltd.
- Hansen G. and Thompson C. (1997) Current Web Architecture (Available from: <http://www.objs.com/survey/WebArch.htm> accessed on: 20/Feb/2009).
- John P. and Quinn L. (2006) Cascading Style Sheets (Available from: <http://htmlhelp.com/reference/css/> accessed on: 30/Nov/2008).
- Kennedy.B, Musciano, *HTML the Definitive Guide*, 3rd Edition, Shroff Publishers and Distributors Pvt. Ltd.
- Purich P. (2006) Understanding the Three-Tier Architecture (Available from: http://www.oracle.com/technology/products/ias/toplink/doc/1013/MAIN/_html/un_dtldev010.htm accessed on: 02/Feb/2009).
- Roger.S Pressman, *SOFTWARE ENGINEERING (A practitioner's Approach)*, 6th edition, McGraw Hill Companies, pp 467-524.
- Sommerville L. (1995), *Software Engineering*, 5th Edition, Addison Wasley Longman (Singapore) Pvt. Ltd.
- Stewart C. (2006) The Advantages of PHP (Available from: <http://www.designersplayground.com/articles/118/1/The-Advantages-of-PHP/Page1.html> accessed on: 10/Dec/2008).
- Suehring S. (2002), *MYSQL (with php, perl, and jdbc/odbc)Bible*, 1st Edition, Wiley Publishing inc. USA.
- Taylor C. (1999) Applying Software Engineering Principles to a Web Site Development Process (Available from: <http://facweb.cs.depaul.edu/ctiphd/ctirs99/online/taylor.html> accessed on: 24/Nov/2008)
- Wampserver Apache, MySQL, PHP on Windows Why PHP 5 Rocks! (Available from: <http://www.wampserver.com/en/presentation.php> accessed on: 15/Dec/2008).
- Webhostinglogic, inc. (2001) Web Database Applications: 3 tier Architecture (Available from: <https://www.webhostinglogic.com/support/support-topics/web-database-3tier.html> accessed on: 04/Mar/2009).
- Wikipedia, the free encyclopedia (2009) Ajax (programming) (Available from: [http://en.wikipedia.org/wiki/Ajax_\(programming\)](http://en.wikipedia.org/wiki/Ajax_(programming)) accessed on: 20/Feb/2009).
- Wood R. (2006) Configuring Your Home Computer to Run Apache Server, PHP, MySQL, ColdFusion, and IIS (Available from: <http://www.tinkertech.net/tutor/wamp/index.html> accessed on: 02/Dec/2008).

Appendix A Interface of the Home Page of “tobemine.com” website

Appendix B Interface of the Registration Page of new user

Appendix C User Profile

Appendix D Service providing part of Home page



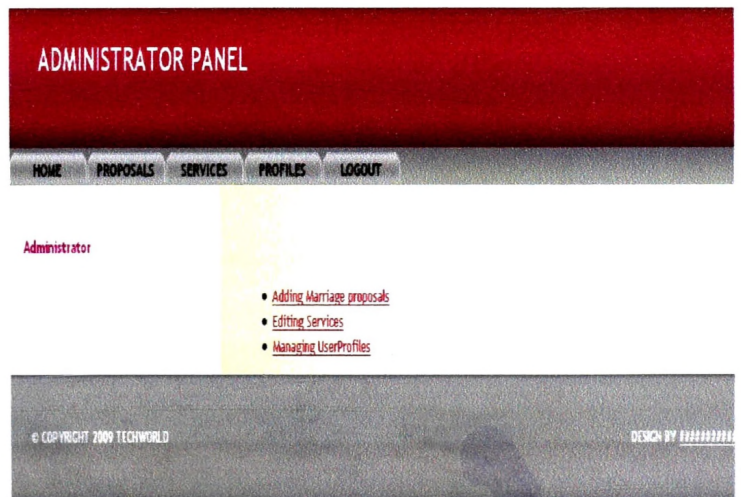
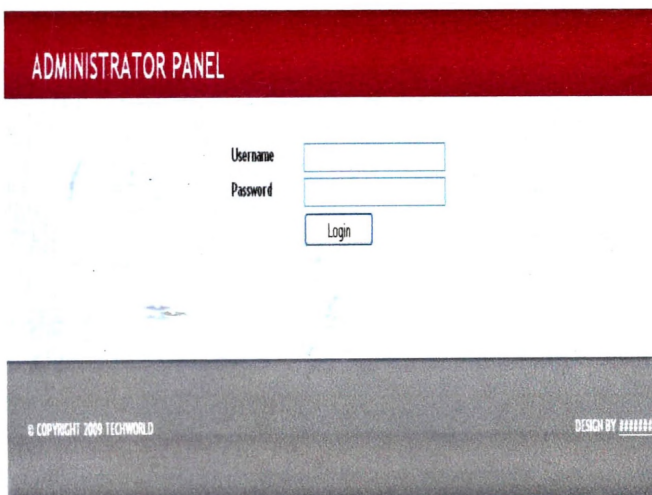
Appendix E View of Members (Search Results)



Partner Search

Seeking a
 With photo
 Of age to
 Of country
 Of religion
 Of community

Appendix F Administrator Panel



Appendix G Messaging System for Users



Profile ID:[upumalee] Logout Settings New messages(0)

Account Search Contact Help

- ♥ Inbox
- ♥ Compose
- ♥ Drafts
- ♥ Sent Messages
- ♥ Trash

Communicate Summary

- **Members awaiting response:** 10 Members
- **Recent messages:** 0 (new) / 2 (old)
- **Accepted Members:** 0 (new) / 4 (old)

Latest Messages

Profile ID	Subject	Date
dryanath	Hai akke	2009-04-02
chathurangi	Hello suku	2009-03-02

[Click here to view all messages >](#)



Profile ID:[upumalee] Logout Settings New messages(0)

Account Search Contact Help

- ♥ Inbox
- ♥ Compose
- ♥ Drafts
- ♥ Sent Messages
- ♥ Trash

Compose Message

To:

Subject:

Message:

B I U Font family Font size

Send Save as a Draft Cancel



Profile ID:[upumalee] Logout Settings New messages(0)

Account Search Contact Help

- ♥ Inbox
- ♥ Compose
- ♥ Drafts
- ♥ Sent Messages
- ♥ Trash

Inbox

Page: 1 of 1

Profile ID	Subject	Date
<input type="checkbox"/> ChamindaSAR	Good luck	27-Mar 09
<input type="checkbox"/> ChamindaSAR	You can chat with me	15-Feb 09

Check All Messages Uncheck All Messages

Forward checked messages to my email address Submit Query

Important Note:
All messages are deleted after 45 days. You can save your messages by forwarding them to your personal email address.

Appendix H Form for Add Wedding Proposals by Administrator

Proposal_ID:
Name:
e-mail:
Contact-No:
Gender: Male
Date Of Birth: Day Month Year
Marital Status: Select
Religion: Select
Country: Select
City:
Community: Select
Height: Select
Complexion: Select
Education: Select
Profession: Select

Description:

Image:
 Please add only JPEG image formats.

Appendix I View oh Wedding Proposals

toBemine.com
Dreaming for Life Partner.....

[HOME](#) [ABOUT US](#) [SERVICES](#) [REGISTER](#) [SEARCH](#) [FAQ](#) [SITE MAP](#) [SHOPPING](#) [LOGIN](#) [CONTACT US](#)

Member Login

Profile ID

Password

[Forget your password?](#)

Partner Search

Seeking a

With photo

Of age to

Of country

Of religion

Of community

[Click here to download plugin.](#)

Success Stories



Ashmi & Manish
{ 815,026 matches }

Miracles do happen
815,026 matches and counting

Hello pathikula.com portal team!!
Accept my thanks for the support I
received in search fo... »

Ms. Ajitha Muthukumarana.

e-mail: ajitha@yahoo.com

Height:

Complexion:

Description: [readmore](#)

tharindu

e-mail: thari@gmail.com

Height: 5ft 7inc - 170cms

Complexion: Jewish

Description: [readmore](#)

Banner Space
Available HERE!

National Digitization Project

National Science Foundation

Institute : Sabaragamuwa University of Sri Lanka

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

2. Date Scanned : ...2017-09-22.....

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

4. Scanning Officer

Name :S.A.C. Gandarwan.....

Signature :.....


Certification of Scanning

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

Certifying Officer

Designation : LIBRARIAN.....

Name : T.N. NEIGHSOOREI.....

Signature :.....

Date : ...2017-09-22.....

Mrs. T.N. NEIGHSOOREI
(MSSc, PGD, ASLA, BA)
Librarian
Sabaragamuwa University of Sri Lanka
P.O. Box 02, Belihuloya, Sri Lanka
Tel: 094 45 2280045
Fax: 094 45 2280045

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