



TURUN AMMATTIKORKEAKOULU
TURKU UNIVERSITY OF APPLIED SCIENCES

Web-Based Library Management System with PHP and MySQL

Bachelor's Thesis

March 2011

Jin Jiawei

Degree Programme in Information Technology

Specialization in Network Communication Technology

BACHELORS'S THESIS | ABSTRACT

Turku University of Applied Sciences

Degree Programme | Information Technology

Date 05.04.2011 | Number of pages 40

Instructor: Yngvar Wikström M. Eng

Jin Jiawei

Web-Based Library Management System with PHP and MySQL

The objective of this Bachelor's thesis was to develop a web-based library management system based on PHP and MySQL in order to reduce the cost of management and make it convenient for the user. The web-based library management system includes the most popular components a common library management system has, administration, book seeker, leasing and E-mail. Besides, it has more humanistic functions such as second-hand online book shop and top 10 ranking. The website was tested on some of the most popular browsers.

The basic functions of Internet Information Services 7.5 and detail features of PHP and MySQL were selected for the purpose of this project. The structure of Library Management System which included a website interacting with a database was determined. The system and development tools were chosen based on their specific features that benefit the system. Furthermore, all necessary components in the library management system were integrated and tested.

The first result of this study was an understanding of the advantages of PHP and how MySQL benefited a large project. The second result was building practical PHP and MySQL projects based on a web server on Windows 7 ultimate.

The understanding of the development tools directly affected the quality of the website. The website which was built in this project offered a high performance, secure, stable and an easy-to-maintain environment. In addition, the website improved the management of a university library. The site was built for testing purposes only.

Key words: library management system, PHP, IIS, MySQL

ACKNOWLEDGEMENT

The thesis was carried out in Turku, Finland. The previous experience in Miccia Oy as a junior web developer supports the practical part of this thesis. I would like to thank my manager Maria Jokelainen for offering the opportunity and help, and my supervisor Yngvar Wikström for guidance and comments on the text and for the valuable advice about implementing the database.

Author's name: Jin Jiawei

Date: 05.04.2011

GLOSSARY AND ABBREVIATIONS

API	Application Programming Interface
ASP	Active Server Pages
CMS	Content Management System
DNS	Domain Name System
ERD	Entity-Relationship Diagram
FTP	File Transfer Protocol
GPL	General Public License
HTML	Hyper Text Markup Language
HTTP	Hypertext Transfer Protocol
IE	Windows Internet Explorer
IIS	Internet Information Services
IMAP	Internet Message Access Protocol
IP	Internet Protocol
IPN	Instant Payment Notification
ISBN	International Standard Book Number
JSP	Java Server Pages
PDT	Payment Data Transfer
PHP	Hypertext Preprocessor

RDBMS	Relational Database Management System
SPARC	Scalable Processor Architecture
SQL	Structured Query Language
TCP	Transmission Control Protocol
UTF-8	Universal Character Set Transformation Format-8 bit
URL	Uniform Resource Locator
WAP	Wireless Application Protocol
WWW	World Wide Web

TABLE OF CONTENTS

ABSTRACT

ACKNOWLEDGEMENT

GLOSSARY AND ABBREVIATIONS

TABLE OF CONTENTS

LIST OF FIGURES

1	INTRODUCTION	1
2	INTERNET INFORMATION SERVICES 7.5 (IIS 7.5)	2
2.1	Features.....	2
2.2	Application pool	3
2.3	Security.....	3
3	HYPertext PREPROCESSOR (PHP)	4
3.1	What is PHP	4
3.2	Installing PHP on Windows.....	5
4	MySQL	6
4.1	What is MySQL.....	6
4.2	Installing MySQL on Windows	7
5	SYSTEM AND DEVELOPMENT TOOLS	9
5.1	Operating System	9
5.2	PhpMyAdmin	9
5.3	EditPlus.....	10
5.4	Dreamweaver	10
6	THE LIBRARY MANAGEMENT SYSTEM	12
6.1	Overview.....	12

6.2	Implementing the Database	14
6.3	Accessing Database	16
6.4	User Authentication.....	17
6.5	Book Searching	20
6.6	Online Shopping	21
6.6.1	Implementing Shopping Cart.....	22
6.6.2	Implementing Payment.....	23
6.7	E-mail Service.....	25
6.8	Top 10 Ranking	27
6.9	Further Functions.....	27
7	Discussion	29
8	Conclusion.....	30

LIST OF FIGURES

Figure 1:	A sample test file	5
Figure 2:	Block diagram of system structure	13
Figure 3:	ER-diagram of the database	14
Figure 4:	The form according to search.php.....	20
Figure 5:	The data flow of the user view in the library management system	21
Figure 6:	Functionality of E-mail Service	25
Figure 7:	Administrator's login form.....	27

1 INTRODUCTION

While seeking for a book in the university's library, students must check what category the book belongs to and check the books within this category one by one. Students might keep the book for a long period; sometimes, remembering to renew the book before a specific date is not very easy; this loan period might exceed the time limit so that the students must pay for a fine and the fine could even be worth as much as to purchasing a second-hand book. Some university libraries have been trying to deal with those problems, Åbo Akademi library, for instance, allows students to search the book from the website and shows the bookshelf location but not the specific row and column and it will remind the student via E-mail, but it doesn't have its own E-mail system.

This thesis was produced by initiating some theoretical ideas from a project in the same field called Koha. Koha provided the method to reduce the cost of maintenance. [1]

The goal of this thesis was to design a website with PHP and MySQL that support the additional functions listed below in addition to the basic functions which are to:

- Display specific location of books in search result
- Support second hand book shop
- Integrate E-mail service and Subscriptions
- Provide Top 10 ranking

This solution was chosen because PHP could be embedded into HTML and its open source itself so that the developer would not need to recode the original source code with any extra funds. Even though the installation phase is discussed, the focus is more on empirical work. Bank negotiation is a part of the shopping cart; however, it is beyond the scope of this thesis. The feature and installation of development tools are introduced. Then the construction of this project is presented and ultimately each component of website is introduced.

2 INTERNET INFORMATION SERVICES 7.5 (IIS 7.5)

Microsoft IIS was used as the web server in this project, even though there were many other web servers, it is the most used web server after Apache and it does not need any complicated configurations before being using [2]. Furthermore, after IIS 6.0 was released, Microsoft's security was greatly improved and especially it cooperated with the .net environment, the cooperation makes the security almost perfect.

2.1 Features

The difficulty of the tool directly affects the amount of users, especially the web publishing tool. After all, many companies want to have their own websites, but do not want to hire a high-paid network administrator to maintain them. Therefore, the companies must use tools which are relatively easy to build the website with.

It is easier to use IIS as the web server, make it work and publish the website and its configuration is graphical.

The language of establishing a website or the forum is varied, such as ASP, PHP, JSP and other languages. It is quite stable to run ASP on IIS, especially in the .net environment. However, PHP is supported on Windows 7 only after a complex configuration.

The FTP Server, Web Management Tools, World Wide Web Services were turned on before the IIS configuration. In order to support PHP, Module Mapping was added in IIS and the executable option associated with the location of php-cgi.exe after installing PHP in the local server.

2.2 Application pool

Application pool was proposed after IIS6.0 was released to support high-performance web application design. It associates one or several applications with one or several groups of process. Those groups can avoid the overflow effectively when the website consumes too large memory. [3]

The website was assigned a separate application pool in this project, an independent process called w3wp.exe was created in the system once the site started, therefore, even if the site hangs up for some reason, it will not influence other sites which ran on the same web server.

If each site used a separate application pool, there would be a problem that the server would be overloaded. Meanwhile, the server would need to allocate more memory and CPU usage. However, for the high performance servers located in a company, using a separate application pool is a best choice to provide the users with the server's strong performance regardless of system resource.

2.3 Security

Timely installation of patches for the IIS web server is especially important since the security patches relate to the system security. The Microsoft official website often publishes the latest security patches. For this reason, the server used in this project was ensured to be updated once a week. And the FTP root directory was assigned to the logical drive in order to prevent hackers' attack. In addition, the FTP default port was modified to other than 21 and the log was enabled in case of server exception.

3 HYPERTEXT PREPROCESSOR (PHP)

3.1 What is PHP

PHP (recursive acronym for PHP: Hypertext Preprocessor) is a widely-used open source general-purpose scripting language that is especially suited for web development and HTML can be embedded into it. The developer could use PHP and HTML to generate the homepage. Once a visitor accesses the index page, the server will execute the PHP command and send the outcomes of implementation to the visitor's browser, however, the difference is that PHP is open source and supports most of the popular platforms and it can be run on a Windows platform and multiple versions of UNIX. [4]

PHP does not require any pre-processing before rapid feedback; it does not need mod_perl to adjust the memory image of server. PHP consumes few resources, as a part of the IIS server, PHP does not need to call an external binary code and the server does not bear any additional burden. In addition to operating the page, PHP can cooperate with HTTP. The cookie and digital signatures management can be modified in the settings, and it provides a good connectivity to the database.

There is no need to have a very special development environment with PHP, the block starts and ends with the tags `<?php` and `?>`. Certainly, PHP could be configured with tags and even in ASP format and it would deal with everything among those signs, but not in the same file.

The PHP programming language is similar to Pascal. There is no need to define any variables before using them, and it is very simple to establish the array and the Hash. PHP also has some object-oriented features which are provided to support and to organizing and packaging the source code.

3.2 Installing PHP on Windows

Since PHP is open source, it is easy to download the Windows Binaries from the PHP official website. The PHP version used in this project is 5.2.0. Above all, the PHP 5.2.0 zip file was downloaded and decompressed to local hard disk. In order to associate PHP with the local database, some extensions were enabled in a text file called php.ini such as mbstring, gd2 and MySQL. In the next place, index.php was added into the default library of IIS so that the server could recognize the php file located in the root of the web server once the visitor accesses the IP address or the DNS name.

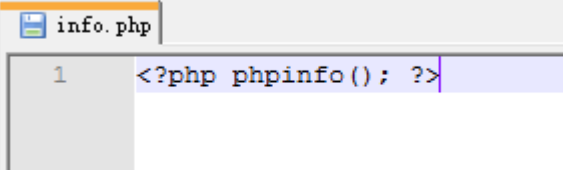
A screenshot of a text editor window. The title bar shows a file named 'info.php'. The editor content shows a single line of code: '<?php phpinfo(); ?>'. The line number '1' is visible on the left side of the editor.

Figure 1. A sample test file.

The successful of installation was tested by creating a file called info.php which is shown in Figure 1, the file was placed into default website folder and the browser showed the result by typing <http://localhost/info.php>. Localhost is the DNS name of local IP address.

4 MySQL

MySQL was owned and sponsored by a Swedish company MySQL AB, now owned by Oracle Corporation [5]. MySQL is free for open source and not-for profit projects. For commercial use, developers have to pay a license fee and paid editions offer additional functionality [6].

4.1 What is MySQL

MySQL is a Relational Database Management System (RDBMS) that runs as a server providing multi-user access to a number of databases. [7]

MySQL was designed for three principles, which are performance, reliability and usability. A cheap, distinctive, fast and efficient RDBMS was created by following those principles. MySQL becomes a perfect tool for developers and administrators to establish maintain and configure complex applications. It has the following main features:

Performance

In the RDBMS, the speed of executing a query and returning the results to the searchers is very important. MySQL is very fast, sometimes the implementation of major orders is even faster than its competitors. The benchmark on the MySQL official website shows that it is superior to almost all other databases such as Microsoft SQL Server 2000 and IBM DB2 [8].

Open Source

The developer of MySQL(MySQL AB) is a strong supporter of open source, and MySQL software could be used smoothly under General Public License (GPL).

Users can download and modify the source code to meet their needs of application, and are free to use it to enhance their applications. [9]

Reliability

In most cases, the higher the performance of the database is, the more it will reduce the reliability. However, MySQL is not the case as it provides maximum reliability and uptime, and a large number of demanding applications has been tested and certified. MySQL's huge user base will help to quickly find and resolve the existent defects, and can test software in a variety of environments; this approach has created almost no defects in the software. In addition, each new version of MySQL must be tested with internal testing and crash-me tool testing, whose main purpose is to reach its limit to access the ability of the system [10].

Portability

MySQL can be run on UNIX and non-UNIX operating systems, including Linux, Solaris, FreeBSD, OS/2, MacOS, and Windows, it can run on a range of architecture, including Intel x86, Alpha, SPARC, PowerPC and IA64, it also supports the 386 series from low to high-end Pentium machines and IBM zSeries mainframes.[11]

4.2 Installing MySQL on Windows

We simply downloaded MySQL windows installer and installed it into local hard disk. The following settings were configured in the MySQL Server:

- Server type
- Database usage
- Path of InnoDB tablespace

- Number of concurrent connections
- TCP/IP Networking
- Server SQL mode
- Default character set
- Windows Service
- Security.

After the configuration of the MySQL Server, the setting was executed into a configuration file. The service was started and the security settings were applied. The default character set used in the server was utf-8. It is widely used transformation format with encoding for the world-wide web and accounting for more than half of all Web pages. [12][13]

5 SYSTEM AND DEVELOPMENT TOOLS

There were four components that needed to be added in the system when the development of the website was made. The computer needed the right kind of operating system to support Microsoft IIS 7.5, the PHP web pages needed to be hosted by the IIS server. Secondly, a database engine and a program were chosen to easily manipulate the construction of the database. The development process was made by using a convenient text editor. At last, the distribution of the website was designed by a web development application.

5.1 Operating System

The programming work was carried out on one computer which ran the Windows 7 Ultimate system. The home version was not chosen since it did not support the IIS web server. The features of IIS 7.5 were introduced in Chapter 2. In the final implementation, the PHP website was tested on two computers which ran Windows 7 and Ubuntu, and it was tested on IE, Chrome, and Firefox.

5.2 PhpMyAdmin

The database management tool used in this project was phpMyAdmin. It is an open source tool written in PHP. It realized the web database management instead of the traditional system implementation.

The following configuration was modified in config.inc.php in order to access the phpMyAdmin from the local host:

```
$cfgServers[1]["host"] = "localhost"; // hostname of MySQL
$cfgServers[1]["port"] = ""; // port of MySQL, default value is 3306
$cfgServers[1]["adv_auth"] = true; // Whether use advanced functions or not
$cfgServers[1]["stduser"] = "username"; // username of administrator
$cfgServers[1]["stdpass"] = "password"; // password of administrator
```

The successful of installation was tested by typing `http://127.0.0.1/phpMyAdmin`.

5.3 EditPlus

The main code of PHP website was written on EditPlus. It is a functional 32-bit compiler which can handle text, HTML and almost all programming languages. The reason why EditPlus was chosen was that it supports HTML, CSS and PHP which were involved in this project.

In addition, it has the following features so that it could make the project development process even faster. It includes:

- fast booting
- support syntax highlighting
- support code completion
- good project management
- built-in Browser.

5.4 Dreamweaver

After compiling the PHP website, Dreamweaver CS5 was used to design the layout of website. After Dreamweaver CS5 was released, Content Management System (CMS) was supported and PHP received better support, not only the programmer could use code hints in the document to search a custom function, but could also use this feature to design better CMS templates. In addition to those features, Dreamweaver CS5 had some other new features, such as its integrated BrowserLab network service that allows the developer to link to a web browser in the laboratory and to using a different browser to check the layout at the same time. In order to test the compatibility of PHP website with different browsers, the

computer needed installations of several browsers, however, CS5 had the build-in Webkit engine included so that it could simulate Safari and Chrome and preview the layout. This did reduce the resources for testing the layout.

Requirements

Since the Dreamweaver was implemented on the Windows system, by way of meeting the needs of stable running, there were some minimum requirements for it and these are:

- Intel® Pentium® 4 or AMD Athlon® 64 processor
- Microsoft® Windows 7® Ultimate or Enterprise
- 512MB of RAM
- 1GB of available hard-disk space for installation
- 1280x800 display with 16-bit video card
- DVD-ROM drive
- Broadband Internet connection required for online services[14].

6 THE LIBRARY MANAGEMENT SYSTEM

The library management system is a web-based widespread information management system which realizes information storage and query. Through the system requires preliminary design, detailed design, coding and testing, the developer could get a good experience of coordination and hands-on development capabilities.

The book search and lending management system which were developed are an important part of the whole library management system, mainly for the retrieval, query and lending of the books in the library. Those are the epitome of the main library management system.

6.1 Overview

The website was used to implement search and book lending management, therefore, the system had the following features:

- The reader could search the books according to the information they have.
- The result is fuzzy-matched when searching.
- The reader could borrow the books based on the searching result they got.
- The reader could check the current lending books and lending history.
- The reader could return the book they are keeping.

The system provided a simple interface for quick book searching, lending and returning. The interface was designed to be mainly used for the common browsers, making the system migration and usage easier.

Structure of Library Management System

Figure 2 shows the whole structure of Library Management System. After a user accessed the IP address of Library Management System, the initialization interface would be shown in the browser unless a server error was detected by the web server. Here, the user could access the top 10 ranking and search modules without login. Once the user had logged in successfully, the user would be able to select the main functions of the Library Management System which were Book Status, Lending, Return Books, Lending Status, Online Shopping, E-mail Service and Logout. If the username and password the user entered did not tally with the values stored in the database server, then the webpage would notice the problem and be turned back to the index page. The web server cooperated with the database server and all the information used in this system was stored in the MySQL database server.

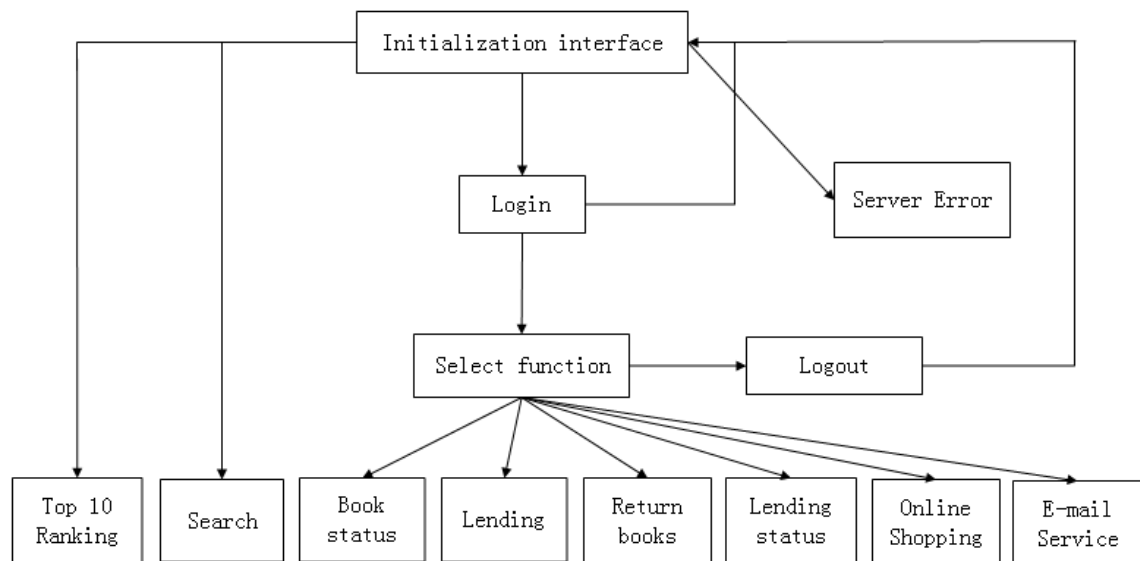


Figure 2. Block diagram of system structure.

6.2 Implementing the Database

In the first phase of system design, we built an ER-diagram for the database. ERD was used to describe information needs and the type of information stored in the database during the requirements analysis phase. Figure 3 shows the ERD of the database. Entity represents a discrete object, it can be considered as a noun, such as Publication, Physical Copy, Location, Loan and Customer. Relationship captures how two or more entities are associated with each other, it can be considered as a verb, such as Copy, Store, Lend and Borrow. Entity sets are drawn as rectangles, relationship sets as diamonds. If an entity set participates in a relationship set, they are connected with a line. Entities and relationships have attributes which are drawn as ovals.

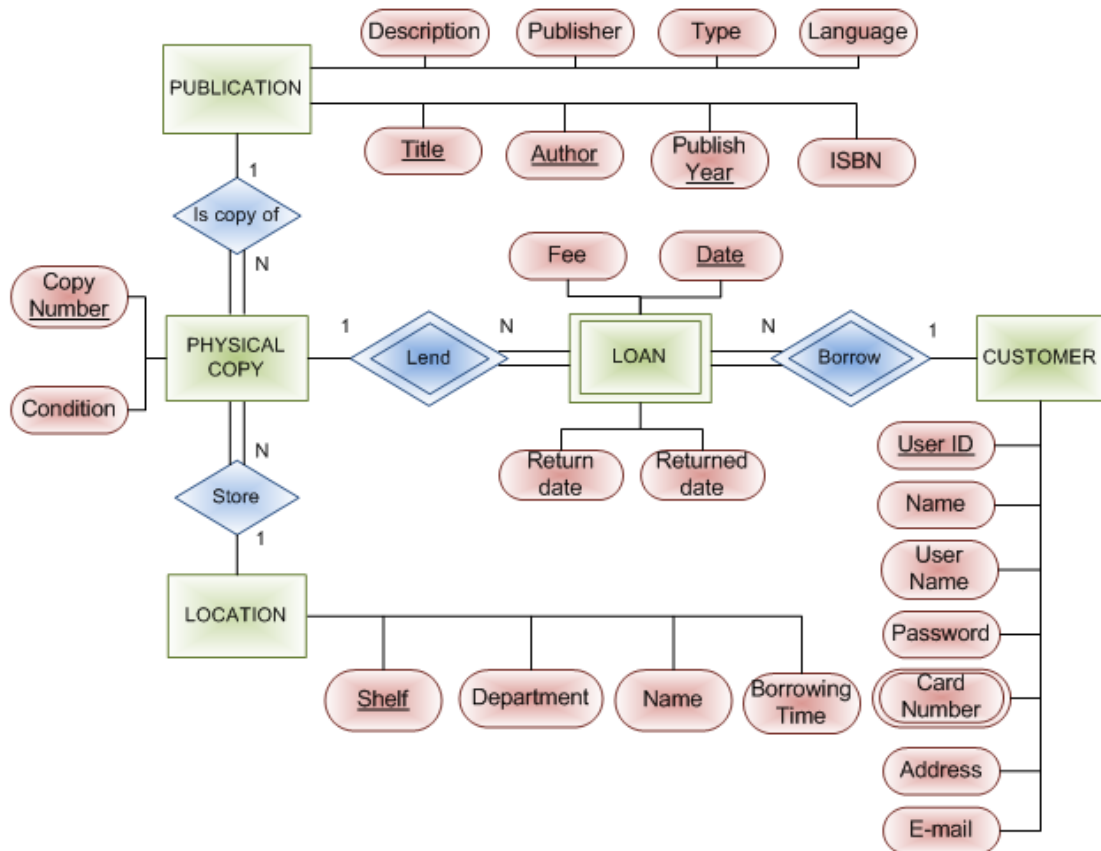


Figure 3. ER-diagram of the database.

Customer

For the library management system to run, we need a customer to either borrow or purchase books. Each customer has a unique number which is User ID; this number is used to identify one customer from the other customers. After a customer registers an account, the customer will hold one personal username and password to access the user authentication. An e-mail message will be created in case of losing the password.

Publication, Physical copy and Location

Publication contains the detailed information of books; each book has a unique title, author and publication year. Other information of the book is needed for the customers to search books. Each publication may have many physical copies and each copy has a copy number which distinguishes it from different copies from the same publication. Shelf, Department and Name are used to define the specific location of a book.

Loan

Loan was created so that the customer may borrow the book he had borrowed again. Once a customer borrows a book, the loan information will be written in the database which contains Data, Return date and User ID. If the customer returns the book later than a specific return date, then a fine will be charged. The cost of the fine will be counted according to the return date and returned date.

Global variables used in the system

- | | |
|--------------------------|-------------------------------------|
| 1. Logged Sign | <code>\$_SESSION["login"]</code> |
| 2. Username | <code>\$_SESSION["username"]</code> |
| 3. Password | <code>\$_SESSION["password"]</code> |
| 4. Connect to SQL Server | <code>@mysql_pconnect()</code> |

5. Select database @mysql_select_db()

The variables above were used in sql_config.php and any other files could use those variables as long as they hold this file.

The relationship between data structures and system

Login module: The system will set the global variables from 1 to 5 after successful login

Log out module: The system will re-set the global variables from 1 to 5 after successful log out.

All other modules need to include the reference to the file sql_config.php to confirm the login information when they are being used. If the visitor is not logged in, the system will turn to the default login page automatically during borrowing and returning books.

6.3 Accessing Database

This system was connected to the database through the specifications of sql_config.php. All modules were required to include sql_config.php when in use, and the syntax of connecting database is as follows:

```
<?php
    $dbhost='localhost';// Database server

    $dbuser='username';//$_SESSION["username");// Database username

    $dbpass="password";//$_SESSION["password");// Database password

    $dbname='library';// Database name

    @mysql_pconnect($dbhost,$dbuser,$dbpass) or die("cannot connect to server!");

    //echo ("config");
```



```
@mysql_select_db($dbname) or die("cannot select database!");  
?>
```

The web server will check the hostname, username and password of database server while connecting to it. After successfully connected, the server will check the name of selected database and search for the values of tables.

6.4 User Authentication

Login and Registration

The first page which was built was called index.php because it provided users with the opportunity to log into the system. In order to accept a user table information and URL parameters, the global variables `$_POST` and `$_GET` were predefined. PHP regular expressions were used to determine whether the user's input tallies with the requirements or not. Session was used to keep the user's login information after detecting user logs. The main function was divided into three parts: user registration, user login and user logout.

User registration

The user registration has four main features which are to:

- fill out the registration information form and javascript detect the initial registration information entered by the user
- check the registration information by registration processing module
- detect whether the username was already exist or not
- write the user's information into the database, registration is successful.

User login

The main features of user login are:

- Javascript detects the initial login information entered by the user in the login form interface.
- The Login module checks the user's input according to the information located in the database server.
- If the information is correct, the user will be notified of successful login and set to login state (session).
- If the information is wrong, the user will be notified of login failure and would need to sign in again.

User logout

The session will be canceled unconditionally if the user logs out.

Creating a database table

According to the system requirements, the fields of the user table are as follows:

Table 1: Table of customer.

Field Name	Field Type	Description
uid	mediumint(8)	Primary key, user's ID, auto increment
name	varchar(20)	User's name
username	char(15)	Registration name
password	char(32)	Password encrypted by MD5
cardnum	tinyint(20)	User's card number
address	varchar(40)	User's address
email	varchar(40)	User's E-mail

Session

Session is a secure dialogue between a web server and a visitor. The session was enabled so that every page within the website could keep this dialogue. For instance, the online shopping cart was a typical session. When the user scheduled a book and added the book into the shopping cart, then the information would be kept. If the chosen book was under the order, the corresponding information would be added into the database; if not, the session would be closed after the user had logged out or closed the browser and the selected book would become void immediately.

The session opens a unique session ID for the user who opened a new session used to identify the user. The session ID would be either stored within the cookie on the user's computer or passed through the URL. Furthermore, the corresponding values of the specific session would be stored in the server which was the main difference with the cookie and the security is relatively higher.

In order to create a session, started with `session_start()` and the system assigned a session ID at the same time. `Session_register()` was used to register a session variable, if this was successful, the system would return TRUE, otherwise return FALSE. Meanwhile, the `session_register()` could register one or more global session variables under the current session, for example:

```
<?php
    session_start();

    $username = "Heidi";

    session_register("username");

?>
```

A variable called username was registered in the session and the value of it is Heidi.

The `session_unregister()` was used to cancel a single session variable, and the `session_unset()` was used to cancel the whole session for example:

```
<?php
    session_start();

    session_unregister("username"); // cancel a variable called username

    session_unset(); //cancel the whole session
?>
```

6.5 Book Searching

This module provided a convenient book searching function, the user could search books based on a variety of conditions. The user should fill out at least one text box so that the system could process the searching function. The system will search a book according to either “and” or “or”, the “exact match” referred to the values stored in the database must be exactly the same as the input and the “fuzzy match” referred to the result could be found as long as the values stored in the database could contain the input.

If the user followed the “Book Searching” menu option, the browser would display the form shown in Figure 4. At this time, the user entered the string and selected the query condition first, then clicked the button “Search” to start searching; the webpage would send a request to the database server through the web server and a value would be returned from the database server.

Title:	<input type="text"/>	Publisher	<input type="text"/>
Author:	<input type="text"/>	Category:	<input type="text"/>
<input type="button" value="Search"/>			
<p style="color: red; font-size: small;">*Please search your books according to one or more terms above.</p>			

Figure 4. The form according to search.php.

This form was generated by the script search.php that supplied a form where users could search their books.

6.6 Online Shopping

Online shopping is the process whereby customers purchase second books from a library directly in the real-time over the Internet. For the buyer, they could purchase a book without an intermediary service as long as they have a computer which is connected to the Internet. For the library, the manager does not need to worry about the stock. Figure 5 shows the user view of library management system which allows users to browse books by category, view book details, add books to the cart and finish the payment.

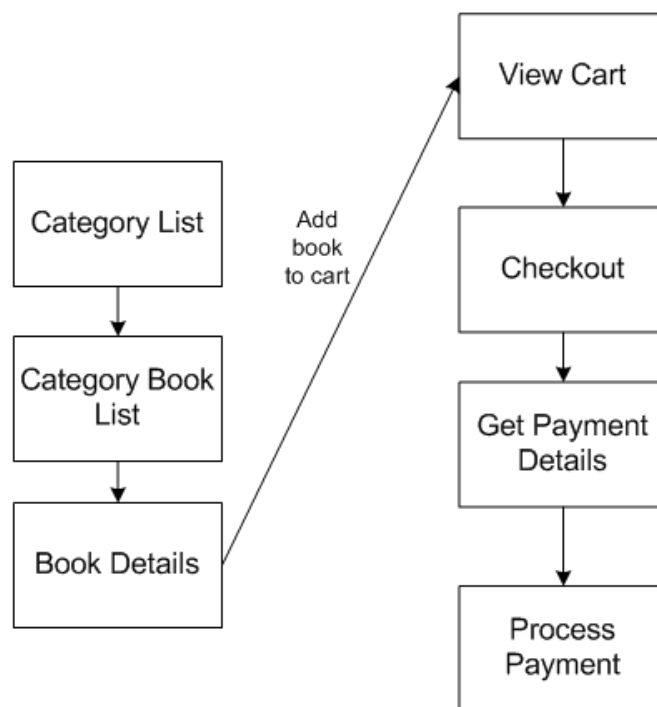


Figure 5. The data flow of the user view in the library management system

6.6.1 Implementing Shopping Cart

In this part, a session variable was used to track user's purchases while they shop. It was easier to use a session variable to track the user's selections because the database was not required to be queried constantly for this information. In addition, the situation that users were just borrowing or changed their minds could be avoided.

Each session was used to distinguish a shopping cart which is equivalent to each of cart's ID number, it also could be considered as a cart's name. If the shopping cart ID and session values did not exist, a new shopping cart would be generated. In order to generate a unique session number, rand() generated a random number first, then uniqid() generated a unique string on the basis of the random number, the string was encrypted by method md5 in the end.

Checking books

When checking the books in the database, the server would check if the "shopping cart" included the "books" in the table or not. If the query failed or the result was not found, the server would return a value 0, otherwise, the server would return a value of the quantity of items.

Adding a new book

When adding an item, the "\$qty = \$this>check_item(\$table, \$session, \$product);" function was used to ensure that such items had been placed in the cart. If the items did not exist, the items would be added into the cart, if so, the quantity of items would be increased based on the quantity of original items and the database would be modified at the same time.

Calculating the amount

When calculating the total amount, each item in the shopping cart was detected separately. If the quantity of items was greater than 0, then the price of each item would be determined and calculated one by one according to the formula: total amount += cost of item * quantity.

6.6.2 Implementing Payment

The payment method used in this system was PayPal. It is an e-commerce business allowing an electronic debit or a credit card payment to be made through the Internet.

In order to implement a PayPal payment, firstly, the web server should hold a PayPal account. The accounts are divided into three types which are Personal, Premier and Business. The main difference among them is the information to be returned for the developer. A Personal user cannot use IPN, PDT and other business tools and the user cannot obtain the transaction information. However, for a Premier or a Business user, IPN and PDT are allowed, after a customer makes the payment, the transaction information will be transferred to a specified website.

Process

After a customer made a payment, PayPal would send a notification to the specific hidden URL of a web server. This notification included the customer's payment information (e.g., name, amount), and a piece of encrypted code. Once the server had received the notification, it would send this information back to the safety PayPal URL. By checking the encrypted string, PayPal could authenticate the transaction. This operation which returned the IPN data back to PayPal prevented a "fraud" so that the server could make sure that the IPN data was sent by PayPal. During verification, PayPal would send the legal confirm information back to the server.

Instant Payment Notification

In order to enable instant payment notification, a URL was defined to receive the notification of user's information. After instant payment notification was enabled, the server would receive a notification once a payment was completed, the notification which hid "From Post" information would be sent to this specific URL and it would contain all payment information. Whenever the server received an IPN from PayPal, the notification would be confirmed before the manager processed

the order. Confirming the information listed would ensure that the transaction is legal.

IPN Confirmation

To ensure that the PayPal account had received the payments, the server must check the E-mail address of "receiver_email" which was verified as registered and confirmed in the PayPal account.

After the server had received the Instant Payment Notification, a confirmation was built to verify that the HTTP POST had been sent to PayPal.

The server sent all incoming variables according to what exact variables had been received. A variable named "cmd" (e.g., cmd = _notify-validate) was attached to the POST string. PayPal would respond to the POST, and the body of the reply would contain a word "VERIFIED" or "INVALID". When the server received a "VERIFIED" notification, the following points were checked before the server implemented the orders, which are to:

- ensure that the status of "payment_status" was "Completed", because the system might send IPN due to other reasons (e.g., Pending, Failed).
- ensure that "txn_id" is not repeated in order to prevent the fraud re-use a completed transaction.
- verify that "receiver_email" had registered an E-mail address in the PayPal account in order to prevent the payment would be sent to fraudster's account.
- check other transaction details such as item number and price and make sure that the price was not changed.

After that, the server would update the database by using IPN data and process purchases. If an "INVALID" notification was received, it would be treated as suspicious.

6.7 E-mail Service

Each user who registered an account in the system would receive an E-mail account. This account was used to remind the user three days before the deadline. If the book was not returned before the deadline, the user would receive a warning every ten days. Further, the user also would get a notification if the books he ordered had been stored in the library. Figure 6 shows mailbox-level functionality and message-level functionality of E-mail service.

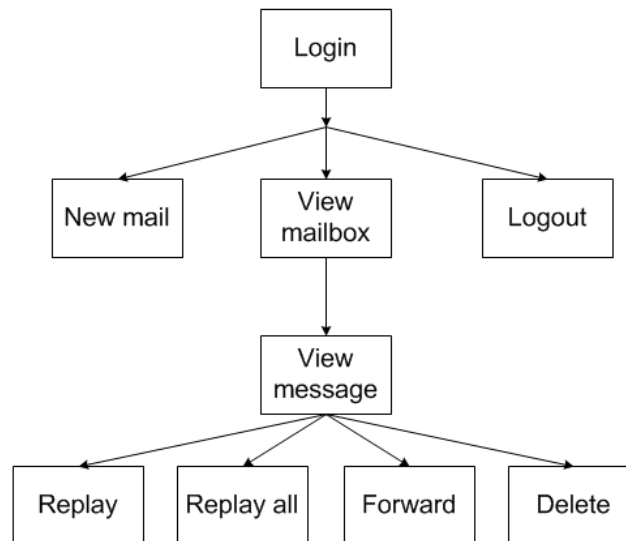


Figure 6. Functionality of E-mail Service.

IMAP Support in PHP

PHP has excellent IMAP support, it is provided via the IMAP function library. Since this system was developed on Windows 7 and the IMAP feature was enabled in the IIS SMTP service, there is no need to install an extra IMAP library. IMAP provides a two-way communication between webmail and the E-mail client; the system will send a feedback of operations on the E-mail client to the server. Meanwhile, a convenient way of downloading the service was provided by IMAP, users could read the mail even if they were offline. An abstract view feature was supported so that users could decide whether they want to download the file or not

after reading all the mails which contain arrival time, subject, sender, size and other information.

Sending Mail

It was the simplest part in the E-mail service. PHP has a ready-made function called `imap_mail` that supports the server to send a mail.

Attachment

Users could attach a file before they sent the mail. The defined variables for the uploaded files are different according to the version of PHP. The automatic global variable `$_FILES` is supported in the current version of PHP. The `$_FILES` array was used to store the information of attached files. The name, type, size, temporary name and error code of the file were defined in the `$_FILES` array.

Once a file was uploaded, it would be saved in a default temporary directory of the server automatically unless the `upload_tmp_dir` was set to another path in the `php.ini` file. The PHP script which accepted uploaded files could decide what it would operate on the files. For instance, the `$_FILES['filename']['size']` variable could be used to ignore the file whose size was too big or too small. No matter how the script operated, the uploaded files would be removed from the temporary directory unless they were moved to another path.

If there was no chosen file in the form, the value of variable `$_FILES['filename']['size']` would be set to 0 and the temporary name would be set to none. If a file was neither moved to another path nor renamed, it would be deleted after the form sent the request.

6.8 Top 10 Ranking

Guests or users could access the top 10 ranking module. After customers were able to view different books, the system would count the watchers once a customer viewed the book. The system ranged the top 10 books every week according to those counts and published the results to the webpage. If a book was sold out or broken, the system would remove it from the top 10 ranking immediately. Those counts were reset monthly since there might be many new books imported into library and the older books hold a large number of counts. The value of counts must be cleared in order to start those counts of both new books and old books at the same point.

6.9 Further Functions

Administration

The Library Management System will integrate an administration module. It is a large background module in the big system. The administrator will have a higher permission account that is able to login to manage the books and users' accounts. The login form for the administrator is shown in Figure 7. The module will have an individual database table to store the information of administrator's account. The Login form was built and the test account for the manager was generated.

>>>Administrator Login	
Username:	<input type="text"/>
Password:	<input type="password"/>
	<input type="button" value="Login"/> <input type="button" value="Reset"/>
Test Account Username: admin Password: admin	

Figure 7. Administrator's login form.

Google Maps Service

This system will integrate a Google Maps service module. Customers will be able to view the location of libraries on the map and this service could route them to the destination based on their location.

There are three different categories of Google Maps API which are JavaScript API, API for Adobe Flash and Static Google Maps API. Static Google Maps API is chosen in this part. It uses an http interface to create pictures of map and does not need JavaScript to support. Further, it could be applied to mobile phone clients and customized clients.

The API Google provided meet the needs of most systems. Although Flex is not a mainstream technology for the development effort, it uses standard web development technology. In addition, it is easy to associate it with the interface of website. Especially within the static maps API, it provides the types of map for WAP used in mobile phones. Besides, the Google Maps API is free for commercial use [15].

Google Map API supports:

- address resolution (resolve from the name of address to specific latitude and longitude)
- zoom level setting
- image size setting
- adding tags
- map route display.

7 Discussion

The site contains important information, such as social identifier, PayPal username and password. If the site does not have proper security mechanisms, it will be hacked in a second day. Therefore, the site should use a web framework like Drupal to prevent three popular hack methods which are SQL injection, cross site scripting and cross site forgery.

SQL injection is an attack which can obtain some secret information in the databases, if we use web framework, all the execution to database will be done through the model so that it protects against SQL injection. Cross site scripting is an attack to which runs a script in our applications, especially if we allow users to post information to our web application. With proper script, attackers can steal session ID of other customers and use them. Cross site forgery is an attack to send HTTP POST to other websites. If some sites like fake PayPal sends fake HTTP POST to our websites and say that "VERIFY", our sites probably will just accept that money. The web framework can reduce this attack.

This system will choose Static Google Maps API to achieve Google Maps service. Although it is the easiest way to show the location of users, Static Google Maps API does not support direction API which means users cannot use navigation service. The sites need to use JavaScript API and have mobile location feature of the browser to achieve navigation service.

8 Conclusion

The main goal of this thesis was to explore the process of building a PHP and MySQL Library Management System. The purpose of the study was to design a user view interface for the Library Management System.

The Library Management System was built on the basis of PHP development; this study reduced the development cost. The further operation and maintenance cost could be gained from advertising fees. As the project grows, more and more libraries will consider purchasing this system. The financial benefit from the system will be much greater than the operation and maintenance costs. Besides, the system used a fully graphical interface, which fully considered the efficiency of managing system.

This thesis was limited since Library Management System will have a manager view interface. The work done in this study is a user view interface of the whole system.

In the latter part of the system development process, the programmer would focus on the combination of the library and website, integrating the News system and administration module. Meanwhile, the maintenance of the system is also very important as well as improving the efficiency of procedures by optimizing the database and simplifying the structure of PHP script.

REFERENCES

- [1] Katipo Communications Ltd. [PDF document]. Available at:
<http://www.parliaments.info/downloads/16%20KOHA%20-%20Presentation.pdf>.
Retrieved 07.07.2006
- [2] NETCRAFT LTD, "Netcraft January 2011 Web Server Survey". [www-document]
available at: <http://news.netcraft.com/archives/2011/01/12/january-2011-web-server-survey-4.html> Retrieved 03.02.2011
- [3] Microsoft MSDN. "What's New in Web Server (IIS) Role in Windows 2008 R2".
[www-document] available at: [http://technet.microsoft.com/en-us/library/dd560629\(W5.10\).aspx](http://technet.microsoft.com/en-us/library/dd560629(W5.10).aspx) Retrieved 25.11.2010
- [4] The PHP Group. [www-document] available at:
<http://www.php.net/manual/en/intro-what-is.php>, Retrieved 04.03.2011
- [5] Sun Microsystems Press, [www-document] available at:
<http://www.sun.com/aboutsun/pr/2008-02/sunflash.20080226.1.xml>, Retrieved
26.02.2008
- [6] Oracle Corporation, "Commercial License for OEM, ISVs and VARs". [www-document] available at: <http://www.mysql.com/about/legal/licensing/oem/> Retrieved
07.2010
- [7] MySQL AB. "What is MySQL?, MySQL 5.1 Reference Manual". [www-document] available at: <http://dev.mysql.com/doc/refman/5.1/en/> Retrieved
19.03.2010
- [8] Hernando, B. *Using IBM DB2 for i as a Storage Engine of MySQL*. Retrieved
18.01.2010
- [9] MySQL AB. "Must I purchase MySQL Enterprise under MySQL's Commercial License in order to receive support?"

- [10] The MySQL Benchmark Suite. [www-document]. Available at: <http://dev.mysql.com/doc/refman/5.1/en/mysql-benchmarks.html>. Retrieved 2011
- [11] Jean-François P., "PCSI Kits of Open Source Software for OpenVMS". [www-document]. Available at: <http://pi-net.dyndns.org/anonymous/kits/> Retrieved 08.06.2009
- [12] Mark D., "Unicode nearing 50% of the web". [www-document]. Available at: <http://googleblog.blogspot.com/2010/01/unicode-nearing-50-of-web.html>. Retrieved 05.12.2010
- [13] W3Techs. "Usage of character encodings for websites". [www-document]. Available at: http://w3techs.com/technologies/overview/character_encoding/all Retrieved 30.03.2010
- [14] Adobe Systems Incorporated. "Adobe Dreamweaver CS5: System Requirements and languages". [www-document]. Available at: <http://www.adobe.com/products/dreamweaver/tech-specs.html?promoid=DRHWV> Retrieved 29.01.2011
- [15] Google, "Google Maps API FAQ". [www-document]. Available at: <http://maps.google.com>