

INTERNATIONAL UNIVERSITY OF SARAJEVO

**Designing and Implementing a Dynamic**

**Website**

**Online Bookstore**

**IBRAHIM GULER**

**1010319**

The Bachelor’s Thesis on Computer Science and Engineering Program

Sarajevo 2016

Approval of the Faculty of Engineering and Natural Sciences

-------------------------

Prof. Dr. Fuat GURCAN

DEAN

I certify that this thesis satisfies all the requirements as a thesis for the degree of Bachelor Of Science in Computer Sciences and Engineering.

-------------------------

Assist. Prof. Dr. Jasminka Hasic Telalovic

Program Coordinator

This is to certify that we have read this thesis and that in our opinion it is fully adequate, in scope and quality, as a thesis for the degree of Bachelor of Computer Sciences and Engineering.

Assist. Prof. Dr. Jasminka [Hasic Telalovic](https://cs.ius.edu.ba/jasminka-hasic-telalovic) -------------------------

Supervisor

**Examining Committee Members (first name belongs to the chairperson of the jury and the second name belongs to supervisor)**

Assist. Prof. Dr. Christian Walter Peter Omlin --------------------

Assist. Prof. Dr. Jasminka Hasic Telalovic -------------------

Assist. Prof. Dr. Emine Yaman --------------------

# DECLARATION

I hereby declare that all information in this document have been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Signature\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Date\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**PREFACE**

I would like to thank my both professors and supervisors Assist.Prof. Dr. Jasminka [Hasic Telalovic](https://cs.ius.edu.ba/jasminka-hasic-telalovic) and Ms. Nesibe Merve Demir for giving valuable comments and reviewing this thesis.

Last but not least,this work would not have been possible without the support and encouragement of my family,colleagues and friends who were always there to support and inspire me towards continuous learning and education.

**ABSTRACT**

The aim of this Bachelor's thesis is to develop an online web application book store to help the users to buy books online. To develop this application few technologies and tools were used. Usages of all these tools assist to construct a desired website. They include PHP, MySQL, HTML, JavaScript, CSS, Bootstrap (to design the interface) and Sublime Text (as a source code editor). It was able to use these tools and technologies by integrating each other to give the desired output. As a result there is a working program that runs in a “localhost” server connection. Users can register and log in to the system in order to interact with the application. As it has not been tested on a real working server on a web, in future the development requires more options for the users for a better interaction with the system in a user friendly and easy way. More technologies might also need to be introduced for a faster and better outcome.

**TABLE OF CONTENTS**

List of Tables ......................................................................................................................... i

List of Pictures ...................................................................................................................... ii

1. INTRODUCTION ............................................................................................................ 11

1.1. Background.................................................................................................................... 11

1.2. Main Goals................... ....................................................................................................11

1.3. Functionalities...................................................................................................................12

1.4. Potential Market User.......................................................................................................12

2. TECHNOLOGY PREVIEW...............................................................................................13

2.1. PHP ..................................................................................................................................13

2.2. MySQL Database ............................................................................................................13

2.3. HTML .............................................................................................................................14

2.4. CSS .................................................................................................................................14

3. REQUIREMENTS ANALYSIS........................................................................................15

3.1.1. Functional Requirements.............................................................................................15

3.1.2. Non Functional Requirements.....................................................................................16

3.2. Use Case Modelling........................................................................................................17

3.2.1. Use Case Diagram for customer...................................................................................17

3.2.2. Use Case Diagram for admin.......................................................................................18

3.3. Detailed Use Case ..........................................................................................................19

3.3.1. Use Case Login Scenerio.............................................................................................19

3.3.2.Use Case Order a book After Login Scenerio………………………………………20

3.3.3. Register Use Case Scenerio.........................................................................................21

3.4. Database Dictionary ......................................................................................................22

4. USER INTERFACE DESIGN .........................................................................................26

4.1 Online Book Store ..............................................................................................................26

4.1.1 Home Page ......................................................................................................................26

4.1.2 Login Page ......................................................................................................................27

4.1.3 Register Page ..................................................................................................................28

4.1.4 Search Page ....................................................................................................................29

4.1.5 Admin Page ....................................................................................................................30

5. CONCLUSION ...................................................................................................................31

REFERENCES .......................................................................................................................32

GLOSSARY............................................................................................................................33 APPENDIX A ........................................................................................................................36

APPENDIX B .......................................................................................................................40

**LIST OF TABLES**

Table-1 Functional Requirements .................................................................................15

Table-2 Nonfunctional Requirements ............................................................................16

Table-3 Login Use Case .................................................................................................19

Table-4 Order a book after log in Use Case ...................................................................20

Table-5 Register Use Case .............................................................................................21

**LIST OF PICTURES**

Picture-1 Use Case Diagram for Customer ............................................................................17

Picture-2 Use Case Diagram for Admin ................................................................................18

Picture-3 User Table ..............................................................................................................22

Picture-4 Book Table .............................................................................................................23

Picture-5 Category Table .......................................................................................................23

Picture-6 Contact Table .........................................................................................................24

Picture-7 Shipping Table .......................................................................................................24

Picture-8 Subcat Table ... .......................................................................................................25

Picture-9 Home Page .............................................................................................................26

Picture-10 Login Page .... .......................................................................................................27

Picture-11 Register Page .......................................................................................................28

Picture-12 Search Page ...........................................................................................................29

Picture-13 Admin Page ...........................................................................................................30

Picture-14 Wamp Server Setup ..............................................................................................45

Picture-15 Installation folder ..................................................................................................46

Picture-16 Wamp PHP mail parameters .................................................................................46

Picture-17 Completing Setup Wizard .....................................................................................47

Picture-18 Localhost ...............................................................................................................47

Picture-19 Starting Wamp Server ...........................................................................................48

**EXPLANATION OF CHARCTERS AND ABBREVIATIONS**

PHP Hypertext Pre-Processor

MySQL A relational database management system based on SQL

SQL Structured Query Language

XML Extensible Markup Language

HTML Hypertext Markup Language

IIS Internet Information Server

URL Uniform Resource Location

HTTP Hypertext Transfer Protocol

DOM Document Object Model

ID Identification

ISBN International Standard Book Number

ER Entity Relation

**1. INTRODUCTION**

**1.1. Background**

With the development and improvement of computer science, internet technology and database technology are used widely. Lot of people choose to shop online instead of walking traditional bookshop. The enterprise IT departments have recognized advantages of the internet. Nowadays, the bookstore website is often used as a platform for selling and purchasing books online. Registered customer can search all books they want to buy.

In general, this application performs the basic registering and logging in of the user and order books with the contact information to the database. Inserting data to the database and fetching it from the database is the prime object of this project. Different database tables were created for specific items to be inserted and to be fetched. Even though this was possible to make, more options and functionalities must be added to make use of the application. For example, so far it is only possible to add items into the cart without purchasing them. It must be enhanced in the future.

During five years of my education, I had already studied some courses like agile development project, programming project and Web Application course including basic HTML, CSS, PHP language, and JavaScript and database system. This is the main reason that made me to develop this project.

**1.2. Main Goals**

The aim is building a web application for users where they can easily register onto the system and browse among hundreds of books with user-friendly interface and be able to buy them online with secure internet payment. This application helps the user to have a wide option and a regular visit to the website.

Our main goal is to give as much functionality as we can for the customers and admins. In addition to this, students can also benefit from this web application where it is difficult to find study books for required courses

**1.3. Functionalities**

This thesis contains a step by step flow of the codes which builds up the entire application. The main purpose is to show how to achieve the online book store system process where customers purchase books and admin management will be introduced. The online bookstore system completes user login, register, browse books, search books, purchase books, staffs manage books information, books categories, orders and accounts.

The declaration and creating of database from the scratch, and also screen shot images of the console which were used in order to build the application will be described in detail. One can use the implementations elsewhere by following the steps and procedures. A special attention must be paid during the formation of files and folders because each functionality of the different files and folders are related to each other since an object oriented programing language has been used. This helps to avoid a nested way of programing style and to make several individual codes of files, which help errors and faults to be traced easily.

**1.4. Intended User and Market Potential**

Today’s advanced technological world with the spectacular growth of internet,new opportunities are available for business across the world which makes now essential to have a web presence by companies. PHP development plays an important role in web development used by the developers to create dynamic web applications in order to reach many people and make their lives easier the worldwide providing useful applications. The online book store web application can be accessed and used by any user even who has minimum technical knowledge about computers as long as an internet access is provided and they have a decent browser running on their computers. It is also very convenient for students to get access to knowledgeable resources.

**2. TECHNOLOGY PREVIEW**

**2.1. PHP**

”Hypertext Preprocessor” is sometime abbreviated to ”PHP”. It is a server-side scripting language and HTML embedded scripting language used to build up dynamic web pages for implementation of web application. PHP is an open source and free download software. Open source means code data have not been encrypted, in order to facilitate people use for the second developing. It usually is used to be share between programmers.

PHP code is interpreted into a normal HTML page content in web server-side, then send it to browser. This way allows us to use it to complete more complex functions.

One major advantage is good cross platform compatibility form nearly all servers (e.g Apache, IIS, etc) and different platforms, like windows, linux and so on.

**2.2. MySQL Database**

MySQL is an open source back end database system. Database system is the main application to store data. It mainly uses in managing the information lists, and the information may be have many different sources, such as commercial transactions, customer requirements, or sales reports and so on. Database system is mainly used for processing information. A good database system can efficiently and quickly process the data rather than data.

SQL is structured query language and it is the easy database language to store data. Also SQL language can update and access the data. In the website, people get a variety of information through the SQL language. For example, books information, customer information, orders information in the online bookstore database system.

**2.3. HTML**

HyperText Markup Language (HTML) is a popular markup language used in web pages. HTML can be simply written in a text editor and tested through web browser. Writing in html is easy; with html it’s also possible to add media and images to the web page. HTML contains special markup tags like <title>, <h>, <p> etc. (Encyclopedia Britannica 2015.) To declare the title of the page for example, the title has to be included in the title tags. Similarly, the paragraphs, headings and other different contents in website pages have to be included inside the respective HTML tags.

It is easy to store HTML code; in a simple text file with filename followed by .html or .htm extension. HTML gives developers the possibility of creating sections in the document. As <title> tag gives the title for the web page, <H1> for example specifies the main content of the web page. Similarly, with H2, H3 and so on tags, HTML creates minor contents. There are tags for other features such as (<p>), style of font (<b>bold</b>) and tables (<table>) etc. (Shannon 2012.)

**2.4. Cascading Style Sheet (CSS)**

CSS is a language that was developed in 1992-1993. CSS helps design each and every element of the markup elements language such as HTML by giving complete control to the designer. While the HTML elements enable the web page designers to add what content they want, CSS makes it possible how to display the content to the user. CSS covers the areas such as colors, layout, advanced positions of elements, fonts and also allows the content to adapt the content to different devices such as phones, tablets, bigger screens and printers. CSS can operate independently as well as be used with any markup languages based in XML. (Lie and Bos 1999.) CSS uses simple, everyday English words and has an easy syntax. CSS is crucial in advanced web designing as it gives control to the layout and offers numerous techniques to make the web page look sophisticated. Currently, the basic features of CSS are supported by all main browsers. CSS has been used for this thesis in the web pages for the development of online book store for positioning, layout, margins and colors for the HTML elements.

**3. REQUIREMENTS ANALYSIS**

This section provides both functional and nonfunctional system requirements list.

**3.1.1 Functional Requirements**

|  |  |  |
| --- | --- | --- |
| **REQUIREMENTS** | **DEFINITION** | **DETAILS** |
| 1.Remote Access | All users shall be able to access the platform | The platform shall be accessible via a browser using internet protocols.The aim is to enable to access and include different types of users anytime,anywhere |
| 2.Accessibility | All users shall be able to access to the interface | The platform interface shall be user friendly |
| 3.Authentication | The system users shall be uniquely identified | Each user shall have a unique username and password linked to the system |
| 4.Information Access | Users with privileges shall be access appropriate information | Users and Admin shall have privileges to access information on their side |
| 5.Editing contents | Already existing books shall be edited or new books shall be added | Updating and removing contents shall be provided by the system |

**Table 1 -Functional Requirements**

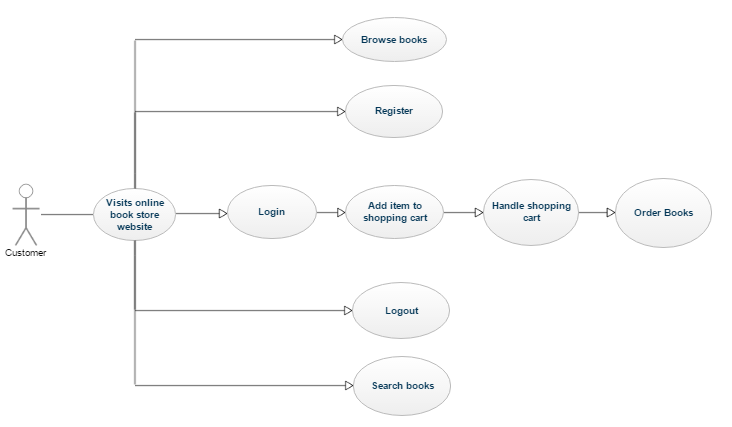
**3.1.2 Non Functional Requirements**

|  |  |  |
| --- | --- | --- |
| **REQUIREMENTS** | **DEFINITION** | **DETAILS** |
| 1.Hardware Configurations | Minimum hardware requirements for server side | **Operating System**: MS Windows all versions, Mac or Linux  **Processor**: Pentium 2.2 MHz or higher  **RAM**:128 MB or more  **Hard disk**:1 GB or more |
| 2.Hardware Configurations II | Minimum hardware requirements for client side | **Operating System**: MS Windows all versions, Mac or Linux  **Processor**: Pentium 2.2 MHz or higher  **RAM**:128 MB or more  **Hard disk**:1 GB or more |
| 3.Software Configuration | Software Requirements for the system | **Application:** PHP/5.5.12  **Web Server**: Apache/2.4.9 **Database:** MySQL 5.6.17 |
| 4.HTML Technology | The system will support HTML 4.01 | The system should run smoothly under IE, Mozilla Firefox, Google Chrome or other common browsers that support HTML 4.01 |
| 5.Performance | Database Performance | Database should perform needed requirements easy and fast |

**Table 2 – Non Functional Requirements**

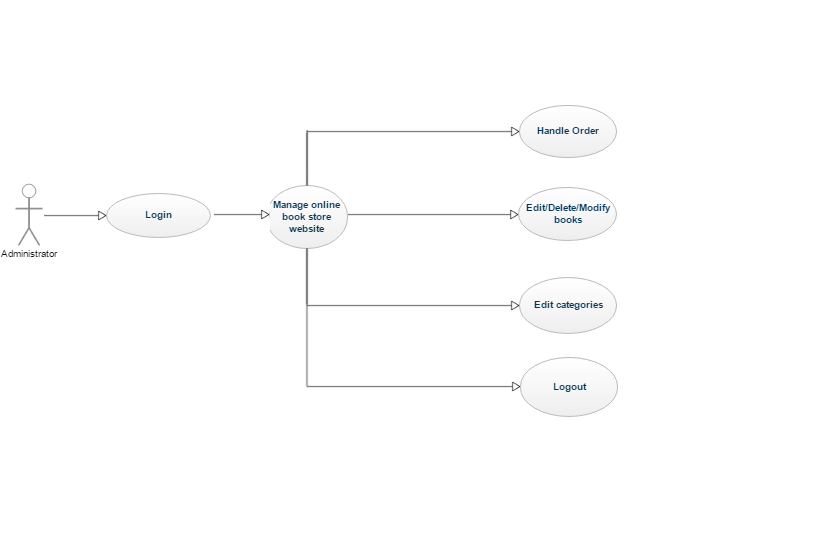
**3.2. Use Case Modeling**

**3.2.1 Use case Diagram for customer**



**Picture 1 – Use case Diagram for Customer**

**3.2.2 Use case diagram for Admin**



**Picture 2-Use Case Diagram for Admin**

**3.3. Detailed Use Case**

**3.3.1 Login Use Case**

Every time user logins the customer or administrator application, he/she must input both of the correct account ID and the password into the input-form. The application will get the input data and send to the System Server, and the server will communicate with the Database and check whether the account ID and password are matched. If it is correct, the application will display the Welcome-Window, and then switch to the Member /admin Interface. Otherwise, the Error-window will be instead.

|  |  |
| --- | --- |
| **Priority** | Essential |
| **Use Frequency** | Always |
| **Direct actor(s)** | Customers, Admin |
| **Pre-condition** | - |
| **Post condition** | The user is logged in |
| **Scenario for log into the system** | |
| **Initial assumptions** | - |
| **Normal** | The user provides a user name and password and proper authentication is applied according to user’s profile |
| **What can go wrong?** | User might enter wrong username/password which can not be recognized by the system.In this case login should be prevented and user should be warned |
| **Other activities** | - |
| **System state on completion** | The user is logged in |

**Table 3-Login use case**

**3.3.2 Order a book after Log in Use Case**

It allows registered users to order books or do changes to the books according to their privileges

|  |  |
| --- | --- |
| **Priority** | Essential |
| **Use Frequency** | Often |
| **Direct actor(s)** | Customers, Admin |
| **Pre-condition** | Login to system as ’customer’ or ‘administrator’ |
| **Post condition** | Add to the shopping cart is opened  Modifying books is opened |
| **Scenario for reaching and changing books after log in** | |
| **Initial assumptions** | Member logged into system |
| **Normal** | User chooses among books and adds it to the cart proceed to payment. Admin can add/remove books |
| **What can go wrong?** | - |
| **Other activities** | - |
| **System state on completion** | The user is logged in |

**Table 4-Order a book after Log in use case**

**3.3.3 Register Use Case**

Firstly, new customers need to register to get one account ID. Information which should be included: customer name, password, email address, phone number and address information. After registration, customers will get an account ID and they can login with account ID and password. The application will insert all the information into the corresponding database tables.

|  |  |
| --- | --- |
| **Priority** | Essential |
| **Use Frequency** | Always |
| **Direct actor(s)** | Customers |
| **Pre-condition** | The user needs to get system code |
| **Post condition** | The user is registered to the system |
| **Scenario for registration to the system** | |
| **Initial assumptions** | - |
| **Normal** | The user fills the registration form and completes registration into the system |
| **What can go wrong?** | The user may not fill all the blanks or fills incorrectly format in the registration form. In this case, user should re-check and try to register again |
| **Other activities** | - |
| **System state on completion** | The user is registered into the system |

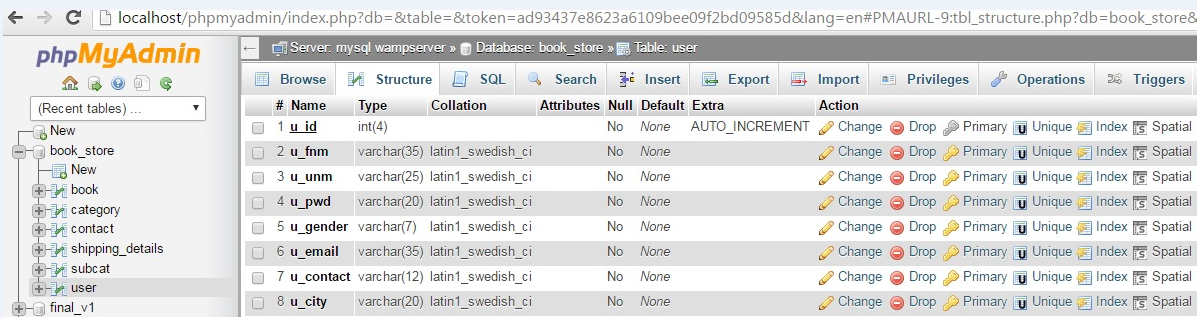
**Table 5-Register Use Case**

**3.4. Database Dictionary**

In this section, the basic structure of the tables composing the database for the project are shown along with information about primary and foreign keys.

**User Table**

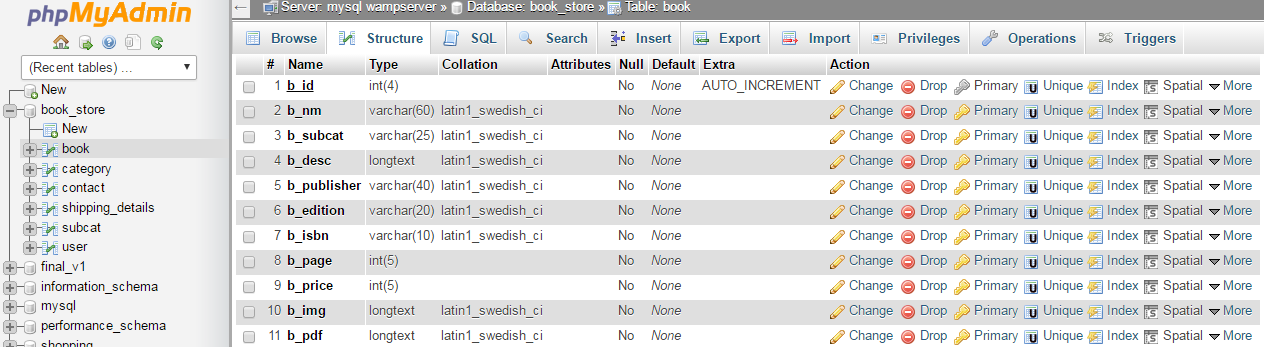
A “user” table which is needed to store all the necessary information about new customers, was designed having the structural contents with eight columns as shown below in Picture 3.

****

**Picture 3- User Table**

**Book Table**

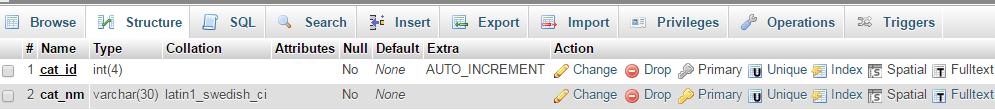
The second table that needed to be created was the “book” table, which is made of eleven columns as shown below in Picture 4.

****

**Picture 4-Book Table**

**Category Table**

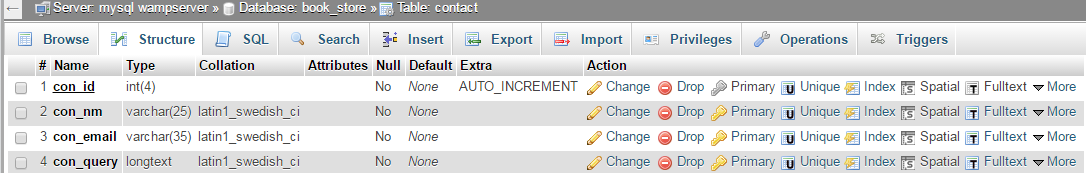
The third table that needed to be created was the “category” table, made of two columns as shown below in Picture 5.

****

**Picture 5- Category Table**

**Contact Table**

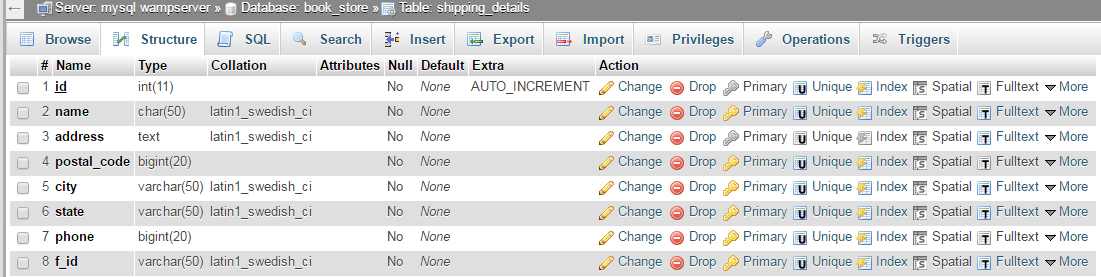
Another table that needed to be created was the “contact” table, made of four columns as shown below in Picture 6.

****

**Picture 6-Contact Table**

**Shipping Table**

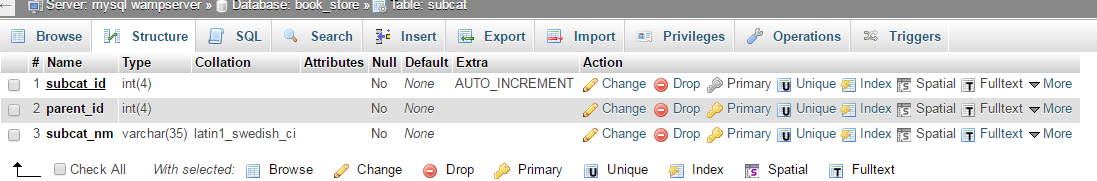
A “shipping” table was designed in order to store the shipping detail contents with eight columns as shown below in Picture 7.

****

**Picture 7-Shipping Table**

**Subcat Table**

The last table for our web software needed was “subcat” to store subcategory contents with three columns as shown below in Picture 8

****

**Picture 8-Subcat Table**

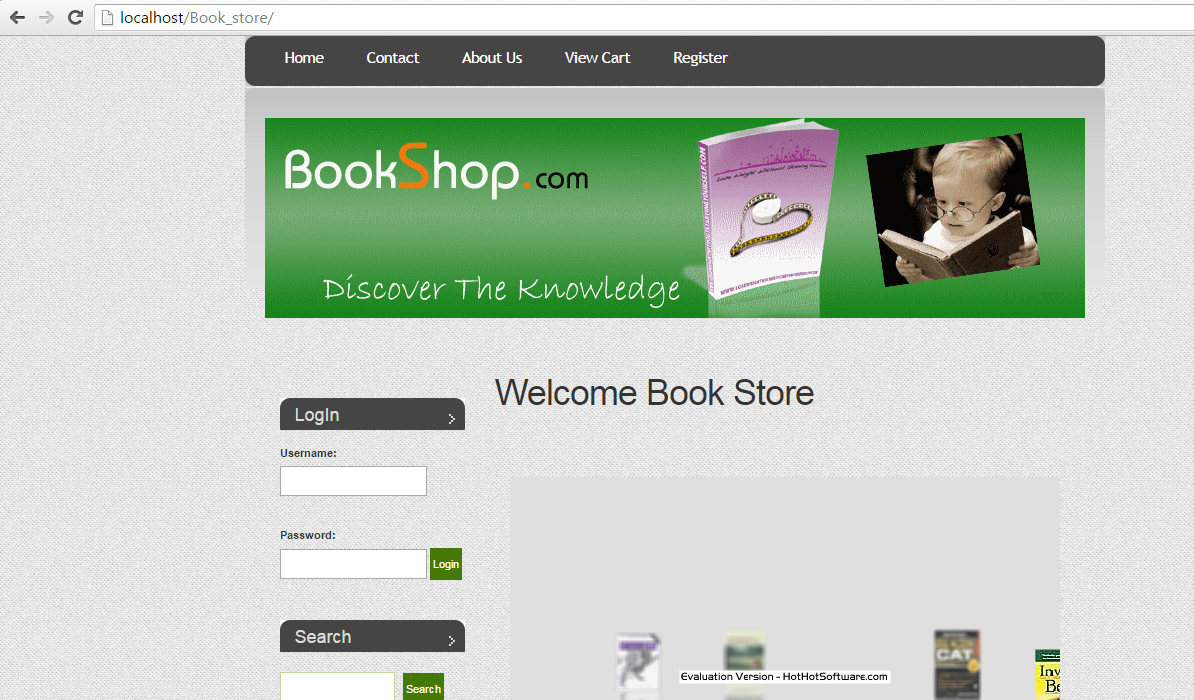
**4. USER INTERFACE DESIGN**

**4.1 Online Book Store System**

The design plan of the application was important for fast and efficient application development. A planned layout of the website was prepared at first. Features such as system design, database design and the way application was intended to work had to be kept in mind before starting the implementation. After downloading the necessary platforms and programs for implementation and learning about the features that were planned to incorporate in the application, the implementation process started.

**4.1.1 Home Page**

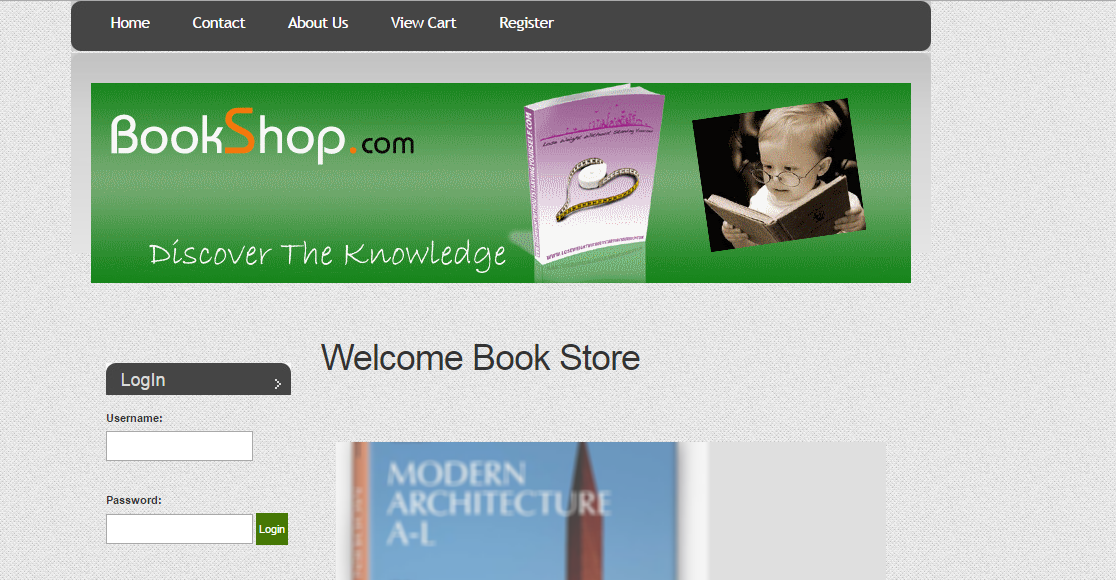
This is the home page of the application when users browse for book store web site

****

**Picture 9-Home Page**

**4.1.2 Login Page**

Login Page requires users to enter their usernames and passwords so that not only allows them to their accounts and continue purchasing books but also allows admin to edit, modify or delete books and see shipments.

****

**Picture 10- Login Page**

**4.1.3 Register Page**

The following picture shows register page of Online book store systems where users need to fill register form to be able to login and view their account. They are asked to enter required data like personal information.

****

**Picture 11- Register Page**

**4.1.4 Search Page**

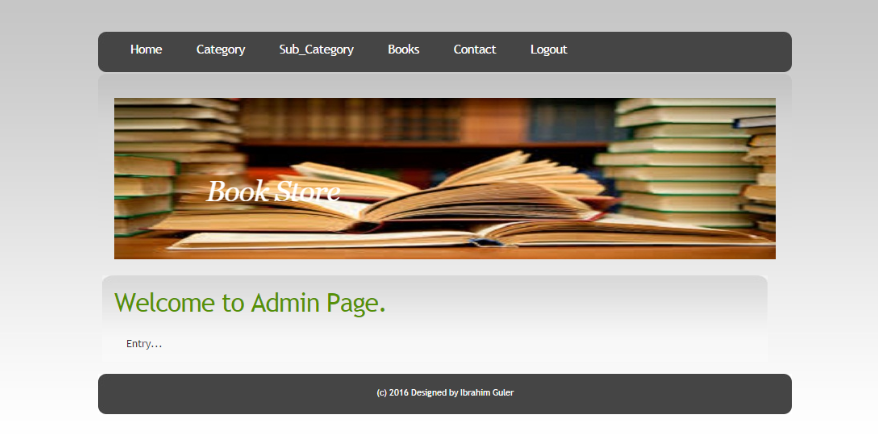
In this function, users can look through the books that are already in the database so they can access them easily.

****

**Picture 12- Search Page**

**4.1.5 Admin Page**

This is the part of view for admin to control over the system where he/she can see the order, modify, add or delete booksand view actions

****

**Picture 13- Admin Page**

**5. CONCLUSION**

After two months of design and development with the guidance of my supervisor, I finally completed the development of my graduation project system based on HTML&CSS, PHP, MySQL and Apache XAMPP.

In design part, I mainly defined the database and initialized the data which has been the best part of having a good understand the database. When I finished the database, I started coding. In this process, some difficulties came up mostly about programming language and connecting with database. I found out relational information on internet and researched open source coding. The problems were solved after I had understood and I modified code in the online bookstore system.

After carefully research online bookstore system, I learned a good professional knowledge to integrate and design of the system. I got a deeper understanding from many of abstract and theoretical knowledge before. At the same time, I had known how to design and development the online bookstore system using the structured programming and models to achieve functions.

However, this project still have some problems that were not being solved or missing parts such as more detailed user accounts. I think it is lack of limited time. I would like to finish and develop this project in a professional way when I have enough time. It is very useful for future work for me.

**REFERENCES**

Build up website, PHP, <http://www.buildwebsite4u.com/advanced/php.shtml> 05.06.2016

Encyclopedia Britannica Ultimate Edition, 2015

Hakon Wium Lie and Bert Bos, *Cascading Style Sheets : Designing for the Web,* 1999

HTML & CSS Design and Build Websites by Jon Duckett

http://php.net/manual/en/book.mysql.php 09.07.2016

http://php.net/manual/en/function.mysql-connect.php 10.07.2016

http://www.agilemodeling.com/artifacts/useCaseDiagram.htm 11.07.2016

http://www.bridging-the-gap.com/functional-specification/ 11.07.2016

http://www.phpknowhow.com/basics/role-of-php-in-web-applications 15.08.2016

http://www.phptherightway.com/ 30.06.2016

http://www.tutorialspoint.com/php/php\_and\_mysql.htm 07.07.2016

http://www.w3schools.com/php/php\_intro.asp 22.06.2016

https://dev.mysql.com/doc/workbench/en/wb-generating-php.html 09.07.2016

https://msdn.microsoft.com/en-us/library/dd409432.aspx 12.08.2016

Mysql, What is Mysql, http://dev.mysql.com/doc/refman/5.0/en/what-is- mysql.html> 20.06.2016

PHP and MYSQL for Dynamic Web Sites by Larry Ullman

Ross Shannon, *What is HTML?,* 2012

**GLOSSARY**

**A**

**Admin:** Administrator of a web system

**Application Server:** An application server is a server computer in a computer network dedicated to running certain software applications.The term also refers to the software installed on such a computer to facilitate the running of other applications

**Authentication:**Process of determining whether someone or something is,in fact,who or what it is declared to be

**B**

**Browser:** A World Wide Web program for navigating the Internet.Most browsers display graphics and formatted pages and let you click to hyperlinks to jump form page to another.Internet Explorer and Google Chrome are browsers.

**C**

**Case Diagram**: A type of behavioral diagram defined by the Unified Modeling Language

**CSS:** A format used to seperate style from structure on web pages.It is a feature of HTML that gives both web developers and users more control over how web pages are displayed.

**D**

**Database:** A database is a collection of information organized into interrelated tables of data and specifications of data objects.

**Database Management System:** A database management system (DBMS) is the software that allows a computer to perform database functions of storing, retrieving, adding, deleting and modifying data. Relational database management systems (RDBMS) implement the relational model of tables and relationships.

**Database Table:** A set of data elements(values) that is organized using model of horizontal rows and vertical columns.

**H**

**HTML:** Hypertext Markup Language (HTML) is a programming language used in the creation of web pages.

**I**

**Implementation**: Is the realization of an application or execution of a plan,idea,model,design,specification,standard,algorithm or policy.In computer science,an implementation is realization of technical specification or algorithm as a program.

**Insert:** The method of inserting bulk data into a table is using the INSERT INTO statement. This allows you to import the results of another SQL statement and is useful for copying full or partial tables.

**Install**: Installing a software program writes the necessary data for running the program on your hard drive.Often the installer program will decompress the data included with the installer immediately before writing the information to your hard drive.

**Interface:** Interface generally refers to an abstraction than an enrty provides of itself to the outside.This seperates the methods of external communication from internal operation and allows it to be internally modified without affecting the way outside entities interact with it,as well as provide multiple abstactions of itself.

**L**

**Login**: In computer security, a log in or log on is the process by which individual access to a computer system is controlled by identification of the user using credentials provided by the user**.**

**M**

**Microsoft Windows**: is an operating systems for personal computers developed and distributed by Microsoft.Windows provides a graphical user interface (GUI),virtual memory management,multitasking and support for many peripheral devices**.**

**MySQL:** is a relational database management system (RDBMS) that runs as a server providing multi-user access to a number of databases**.**

**N**

**Network:** A collection of terminals,computers,and other equipment that use communication channels to share data

**O**

**Operating system:** the low-level software that supports a computer's basic functions, such as scheduling tasks and controlling peripherals

**P**

**PHP:** The PHP Hypertext Preprocessor allows web developers to create dynamic content that interacts with databases.PHP code is embedded into the HTML source document and interpreted by a web server with a PHP processor module which generates the web page document.

**Primary key:**Candidate key to uniquely identify each row in a table.

**Priority:** The fact or condition of being regarded or treated as more important than others.

**Privilege:** Special individual right not enjoyed by others , benefits given only to people of a particular group

**R**

**RAM:** RAM is an acronym for random access memory, a type of computer memory that can be accessed randomly; that is, any byte of memory can be accessed without touching the preceding bytes. RAM is the most common type of memory found in computers and other devices, such as printers.

**Remote Access:** Remote access refers to the ability to access a computer, such as a home computer or an office network computer, from a remote location

**APPENDIX A**

**SOURCE CODE**

**Connection to Database Page**

The following code establishes the connection with the database :

<?php

/ \*\*

connecting to the database

\*/

$conn=mysqli\_connect("localhost","root","","book\_store")or die("Can't Connect...");

?>

**Login Page**

process\_login.php

<?php session\_start();

require('includes/config.php');

if(!empty($\_POST))

{ $msg="";

if(empty($\_POST['usernm']))

{

$msg[]="No such User";

}

if(empty($\_POST['pwd']))

{

$msg[]="Password Incorrect........";

}

if(!empty($msg))

{

echo '<b>Error:-</b><br>';

foreach($msg as $k)

{

echo '<li>'.$k;

}

}

else

{

$unm=$\_POST['usernm'];

$q="select \* from user where u\_unm='$unm'";

$res=mysqli\_query($conn,$q) or die("wrong query");

row=mysqli\_fetch\_assoc($res);

if(!empty($row)){

if($\_POST['pwd']==$row['u\_pwd']){

$\_SESSION=array();

$\_SESSION['unm']=$row['u\_unm'];

$\_SESSION['uid']=$row['u\_pwd'];

$\_SESSION['status']=true;

if($\_SESSION['unm']!="admin")

{

header("location:index.php");}

else {

header("location:admin/index.php");}}

else

{

echo 'Incorrect Password....';}}

else {

echo 'Invalid User';}}

}

else

{

header("location:index.php");

}

?>

**Register Page**

**register.php**

<?php session\_start(); ?>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<?php

include("includes/head.inc.php");

?>

</head>

<body>

<!-- start header -->

<div id="header">

<div id="menu">

<?php

include("includes/menu.inc.php");

?>

</div>

</div>

<div id="logo-wrap">

<div id="logo">

<?php

include("includes/logo.inc.php");

?>

</div> </div>

<div id="page">

<!-- start content -->

<div id="content">

<div class="post">

<h1 class="title">Welcome to Registeration.</h1>

<div class="entry">

<br><br>

<?php

if(isset($\_GET['error']))

{

echo '<font color="red">'.$\_GET['error'].'</font>';

echo '<br><br>';

}

if(isset($\_GET['ok']))

{

echo '<font color="blue">You are successfully Registered..</font>';

echo '<br><br>';

}

?>

<table>

<form action="process\_register.php" method="POST">

<tr>

<td><b>Full Name :</b>&nbsp;&nbsp;</td>

<td><input type='text' size="30" maxlength="30" name='fnm'></td>

**register\_process.php**

<?php

require('includes/config.php');

if(!empty($\_POST))

{

$msg="";

if(empty($\_POST['fnm']) || empty($\_POST['unm']) || empty($\_POST['gender']) || empty($\_POST['pwd']) || empty($\_POST['cpwd']) || empty($\_POST['mail'])||empty($\_POST['city']))

{

$msg.="<li>Please full fill all requirement";

}

if($\_POST['pwd']!=$\_POST['cpwd'])

{

$msg.="<li>Please Enter your Password Again.....";

}

if(!ereg("^[a-z0-9\_]+[a-z0-9\_.]\*@[a-z0-9\_-]+[a-z0-9\_.-]\*\.[a-z]{2,5}$",$\_POST['mail']))

{

$msg.="<li>Please Enter Your Valid Email Address...";

}

if(strlen($\_POST['pwd'])>10)

{

$query="insert into user(u\_fnm,u\_unm,u\_pwd,u\_gender,u\_email,u\_contact,u\_city)

values('$fnm','$unm','$pwd','$gender','$email','$contact','$city')";

mysqli\_query($conn,$query) or die("Can't Execute Query...");

header("location:register.php?ok=1");

}

}

else

{

header("location:index.php");

}

?>

$msg.="<li>Please Enter Your Password in limited Format....";}

if(is\_numeric($\_POST['fnm'])){

$msg.="<li>Name must be in String Format...";}

if($msg!=""){

header("location:register.php?error=".$msg);

}

else{

$fnm=$\_POST['fnm'];

$unm=$\_POST['unm'];

$pwd=$\_POST['pwd'];

$gender=$\_POST['gender'];

$email=$\_POST['mail'];

$contact=$\_POST['contact'];

$city=$\_POST['city'];

$query="insert into user(u\_fnm,u\_unm,u\_pwd,u\_gender,u\_email,u\_contact,u\_city)

values('$fnm','$unm','$pwd','$gender','$email','$contact','$city')";

mysqli\_query($conn,$query) or die("Can't Execute Query...");

header("location:register.php?ok=1");

}

}

else

{

header("location:index.php"); }

?>

**Search.php**

<li id="search">

<h2>Search</h2>

<form method="GET" action="search\_result.php">

<fieldset>

<input type="text" id="s" name="s" value="" />

<input type="submit" id="x" value="Search" />

</fieldset>

</form>

</li><li>

<?php

$query="select \* from category ";

$res=mysqli\_query($conn,$query);

while($row=mysqli\_fetch\_assoc($res))

{

echo'<li><a href="subcat.php?cat='.$row['cat\_id'].'&catnm='.$row["cat\_nm"].'">'.$row["cat\_nm"].'

</a></li>';

}

mysqli\_close($conn);

$search=$\_GET['s'];

$query="select \*from book where b\_nm like '%$search%'";

$res=mysqli\_query($conn,$query) or die("Can't Execute Query...");

**Viewcart.php**

<?php session\_start();

require('includes/config.php');

?>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<?php

include("includes/head.inc.php");

?>

</head>

<body>

<!-- start header -->

<div id="header">

<div id="menu">

<?php

include("includes/menu.inc.php");

?>

</div>

</div>

<div id="logo-wrap">

<div id="logo">

<?php

include("includes/logo.inc.php");

?>

</div>

</div>

<?php

$tot = 0;

$i = 1;

if(isset($\_SESSION['cart']))

{

foreach($\_SESSION['cart'] as $id=>$x)

{

echo '

<tr>

<Td> '.$i.'

<td> '.$x['cat'].'

<td> '.$x['nm'].'

<td> <input type="text" size="2" value="'.$x['qty'].'" name="'.$id.'">

<td> '.$x['rate'].'

<td> '.($x['qty']\*$x['rate']).'

<td> <a href="process\_cart.php?id='.$id.'">Delete</a>

</tr>

';

$tot = $tot + ($x['qty']\*$x['rate']);

$i++;

}

}

?>

<tr><td colspan="7"><hr style="border:1px Solid #a1a1a1;"></tr>

<tr>

<td colspan="6" align="right">

<h4>Total:</h4>

</td>

<td> <h4><?php echo $tot; ?> </h4></td>

</tr>

<tr><td colspan="7"><hr style="border:1px Solid #a1a1a1;"></tr>

<Br>

</table>

<br><br>

<center>

<input type="submit" value=" Re-Calculate " >

<a href="checkout.php">CONFIRM & PROCEED<a/>

</center>

</form>

</div>

</div>

</div>

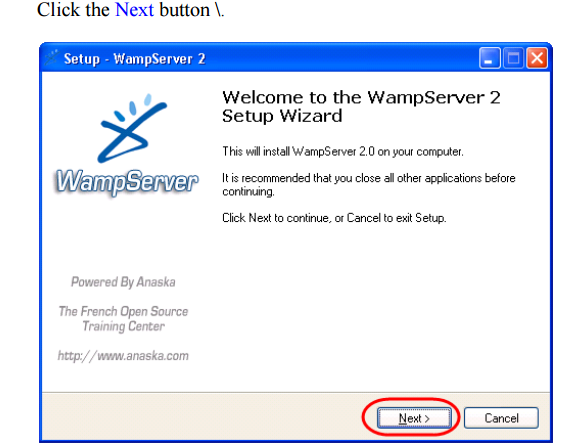
**APPENDIX B**

**INSTRUCTION MANUAL**

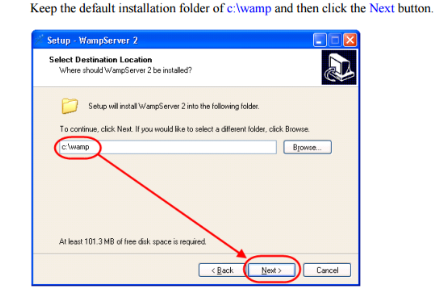
**1.0 XAMP installation**

First Steps:

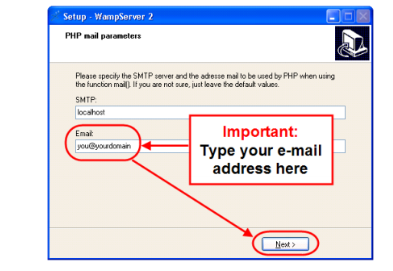
* Install file should been downloaded from “www.wampserver.com/en/#download-wrapper”
* Run “WAMP-WIN64-2.5-INSTALLER.EXE”



Picture-14 WampServer Setup



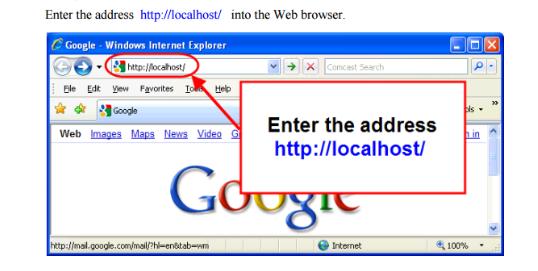
Picture-15 Wamp Server Installlation folder



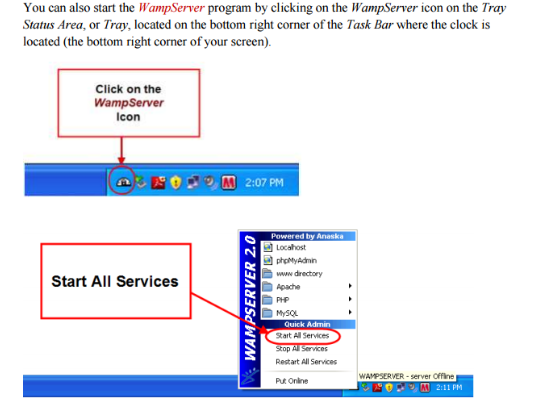
Picture-16 Wamp PHP mail parameters



Picture-17 Completing Setup Wizard



Picture-18 Localhost



Picture-19 Starting WampServer