**LIBRARY MANAGEMENT SYSTEM:**

INTRODUCTION:

Library management System (LMS) provides a simple GUI (graphical user interface) for the Library Staff to manage the functions of the library effectively. Usually when a book is returned or issued, it is noted down in a register after which data entry is done to update the status of the books in amoderate scale. This process takes some time and proper updation cannot be guaranteed. Such anomalies in the updation process can cause loss of books. So a more user friendly interface which could update the database instantly, has a great demand in libraries.

The goal of this project is to provide simplicity as well as security and efficiency to the management. Its main objective is to provide all the functions of the library along with support of barcode reader. The contemporary system stores the database information on a local computer,

which cant be updated and accessed on a remote computer or a computer in

the library network.

**8.1 PROBLEM DESCRIPTION:**

* The Library Management System implements databases to make the existing system more efficient.
* It is difficult to catch defaulters in a usual library system, but LMS solves this problem by providing messages to the administrator about the fine to be paid and books to be returned.
* When a book is to be borrowed, its barcode is read and is fed to LMS.
* When the administrator tries to issue a book to a member, LMS checks whether the member is allowed to borrow books respective to its limit.
* LMS also checks whether the book to be issued is a reference book or not. If the book is a reference book, it cannot be issued
* If the book has no restrictions it is issued to the member, and also the
* members details is updated with
* the books that he has just borrowed.
* Any member can search a book with its title name, or author’s name.

**8.2 DETAILED PROJECT OVERVIEW**

**EXISTING SYSTEM:**

In the existing system, the transactions of the books are recorded in alocal computer and the information cannot be accessed by other members of the library

In some libraries, the transactions are recorded in a register, and then later they are fed into the computer by a Data Entry Operator.

This type of maintenance usually causes some errors due to delay in updation and mistakes during large data entry.

Basically there is no security, i.e. any person can perform transactions other than the administrator.

**PROPOSED SYSTEM**

In Library Management System, the processing of transactions is done with the help of computer.

LMS provides a very simple GUI.

Through LMS, updating to the database is very efficient and instantly done.

It provides a guest login for the members to view details about the available books and allows them to search for a particular book.

LMS provides security (i.e. only the administrator can update any information to the database

Barcode Readers can be installed to the system and respective plug-in can be installed to the LMS for effective and easy transactions.

 Error generation for Authentication purpose and also for constraints to the members on the number of books allowed for a particular member.

 Member can be a Student, or Lecturer or Non teaching Staff, and respective limits for borrowing books for each type of member is set(i.e. students can borrow up to 3 books and lecturers & non-teaching staff can borrow up to 5 books).

According to update anomalies, proper errors are generated in exclamation and dialog boxes.

 The data can be updated to a server on the network. There can be many clients around the library from which the users can check for books available.

**SYSTEM DESIGN:**

LMS implements two types of users in the system. One is theadministrator who has the total control over the database and the other is theguest account through which one can view the READ ONLY terminals of the database.

**Functions given to the Administrator:**

**Administration:**

**a)Adding a new member:**

The administrator can add new members to the library. He has to enter thefollowing details.

i)Patron ID

ii)Patron Name

iii)Designation (Student, Lecturer, Non Teaching staff)

iv)Date of Joining

v)Date of Expiry of membershipA unique patron id is given to each member of the library

**b)Modifying the member details:**

The administrator has the option to modify the details of a member

i)Name

ii)Designation

iii)Imposing or Clearing Fine

iv)Renewing the membership

**c)Removing a member**

The administrator has the rights to remove a member from the library. Most

importantly LMS doesn‟t allow a member to be removed from the library if the

member has books to return or fine to be paid.The member can be removed either with reference to his name or his patron idwhich can also be scanned from his ID card through a barcode reader.

**2.Tech Processing**

**a)Primary Catalogue**

This menu gives options to the administrator to add new books to the library The following details of the book are to be entered.

i) Item Barcode

ii)Title

iii)Author

iv)Physical Description (no. of pages)

v)Reference or not (Reference or Ordinary)

vi)ISBN

**b)Modify Catalogue**

This form displays the all the books of the library and gives the option to the administrator to modify the details of a book. The following details can be modified.

i)Title

ii)Author

iii)Reference or not

iv)ISBN

v)Physical Description (no. of pages)

The barcode can’t be modified.

**3.Circulation**

**a)Check In**

By check in option, a book can be returned to the library. The barcode of the book is required, which can be scanned by the barcode reader.

**b)Check Out**

When a book is check out request is given, LMS verifies whether the person is allowed to borrow more books or not. A student can borrow only up to 3books, whereas a Lecturer or Non teaching Staff can borrow up to 5 books. An error is generated if the person is not allowed.

**4.Queries:**

**a)Search by**

i)Title

ii)Item Barcode

iii)Patron ID This displays the books borrowed by the particular Member with reference to his Patron ID

iv)Patron Name This displays the books borrowed by the particular member with reference to his name

v)Books out Displays the books which are out of the library.

vi)Books available Displays the books which are available for checking out.

**Functions given to the Guest User**

**1)Queries**

**a)Search by**

i)Title

ii)Item Barcode

iii)Patron ID This displays the books borrowed by the particular Member with reference to his Patron ID

iv)Patron Name This displays the books borrowed by the particular member with reference to his name

v)Books out

Displays the books which are out of the library.vi)

Books available Displays the books which are available for checking out.

8.3 FOSS TOOLS USED FOR PROJECT DEVELOPMENT:

1.HTML 5:

Html 5 is a markup language used for structuring and presenting content on the World Wide Web. It is the fifth and current major version of the HTML standard. With Cascading Style Sheets (CSS) and JavaScript, it forms a triad of cornerstone technologies for the World Wide Web.Web browsers receive HTML documents from a web server or from local storage and render the documents into multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document. HTML can embed programs written in a scripting language such as JavaScript, which affects the behavior and content of web pages. Inclusion of CSS defines the look and layout of content. The World Wide Web Consortium (W3C), maintainer of both the HTML and the CSS standards, has encouraged the use of CSS over explicit presentational HTML since 1997.

History:

In 1980, physicist Tim Berners-Lee, a contractor at CERN, proposed and prototyped ENQUIRE, a system for CERN researchers to use and share documents. In 1989, Berners-Lee wrote a memo proposing an Internet-based hypertext system.[6] Berners-Lee specified HTML and wrote the browser and server software in late 1990. That year, Berners-Lee and CERN data systems engineer Robert Cailliau collaborated on a joint request for funding, but the project was not formally adopted by CERN. In his personal notes[7] from 1990 he listed[8] "some of the many areas in which hypertext is used" and put an encyclopedia first.The first publicly available description of HTML was a document called "HTML Tags", first mentioned on the Internet by Tim Berners-Lee in late 1991.[9][10] It describes 18 elements comprising the initial, relatively simple design of HTML. Except for the hyperlink tag, these were strongly influenced by SGMLguid, an in-house Standard Generalized Markup Language (SGML)-based documentation format at CERN. Eleven of these elements still exist in HTML 4.

2. CSS3:

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language like HTML.[1] CSS is a cornerstone technology of the World Wide Web, alongside HTML and JavaScript.

CSS is designed to enable the separation of presentation and content, including layout, colors, and fonts.[3] This separation can improve content accessibility, provide more flexibility and control in the specification of presentation characteristics, enable multiple web pages to share formatting by specifying the relevant CSS in a separate .css file, and reduce complexity and repetition in the structural content.Separation of formatting and content also makes it feasible to present the same markup page in different styles for different rendering methods, such as on-screen, in print, by voice (via speech-based browser or screen reader), and on Braille-based tactile devices. CSS also has rules for alternate formatting if the content is accessed on a mobile device.

3. Javascript:

JavaScript often abbreviated as JS, is a high-level, interpreted programming language. It is a language which is also characterized as dynamic, weakly typed, prototype-based and multi-paradigm.

Alongside HTML and CSS, JavaScript is one of the three core technologies of the World Wide Web. JavaScript enables interactive web pages and thus is an essential part of web applications. The vast majority of websites use it, and all major web browsers have a dedicated JavaScript engine to execute it. As a multi-paradigm language, JavaScript supports event-driven, functional, and imperative (including object-oriented and prototype-based) programming styles. It has an API for working with text, arrays, dates, regular expressions, and basic manipulation of the DOM, but the language itself does not include any I/O, such as networking, storage, or graphics facilities, relying for these upon the host environment in which it is embedded .Initially only implemented client-side in web browsers, JavaScript engines are now embedded in many other types of host software, including server-side in web servers and databases, and in non-web programs such as word processors and PDF software, and in runtime environments that make JavaScript available for writing mobile and desktop applications, including desktop widgets.

4. PHP 7.01:

PHP: Hypertext Preprocessor (or simply PHP) is a server-side scripting language designed for web development but also used as a general-purpose programming language. It was originally created by Rasmus Lerdorf in 1994, the PHP reference implementation is now produced by The PHP Group. PHP originally stood for Personal Home Page,[3] but it now stands for the recursive acronym PHP: Hypertext Preprocessor.

PHP code may be embedded into HTML code, or it can be used in combination with various web template systems, web content management systems, and web frameworks. PHP code is usually processed by a PHP interpreter implemented as a module in the web server or as a Common Gateway Interface (CGI) executable. The web server combines the results of the interpreted and executed PHP code, which may be any type of data, including images, with the generated web page. PHP code may also be executed with a command-line interface (CLI) and can be used to implement standalone graphical applications. The standard PHP interpreter, powered by the Zend Engine, is free software released under the PHP License. PHP has been widely ported and can be deployed on most web servers on almost every operating system and platform, free of charge.

5. MYSQL:

MySQL is an open-source relational database management system (RDBMS). name is a combination of "My", the name of co-founder Michael Widenius's daughter, and "SQL", the abbreviation for Structured Query Language. The MySQL development project has made its source code available under the terms of the GNU General Public License, as well as under a variety of proprietary agreements. MySQL was owned and sponsored by a single for-profit firm, the Swedish company MySQL AB, now owned by Oracle Corporation.[8] For proprietary use, several paid editions are available, and offer additional functionality. MySQL is a central component of the LAMP open-source web application software stack (and other "AMP" stacks). LAMP is an acronym for "Linux, Apache, MySQL, Perl/PHP/Python". Applications that use the MySQL database include: TYPO3, MODx, Joomla, WordPress, Simple Machines Forum, phpBB, MyBB, and Drupal. MySQL is also used in many high-profile, large-scale websites, including Google (though not for searches), Facebook, Twitter, Flickr, and YouTube.

6. Xampp

XAMPP is a free and open source cross-platform web server solution stack package developed by Apache Friends,[2] consisting mainly of the Apache HTTP Server, MariaDB database, and interpreters for scripts written in the PHP and Perl programming languages. XAMPP stands for Cross-Platform (X), Apache (A), MariaDB (M), PHP (P) and Perl (P). It is a simple, lightweight Apache distribution that makes it extremely easy for developers to create a local web server for testing and deployment purposes. Everything needed to set up a web server – server application (Apache), database (MariaDB), and scripting language (PHP) – is included in an extractable file. XAMPP is also cross-platform, which means it works equally well on Linux, Mac and Windows. Since most actual web server deployments use the same components as XAMPP, it makes transitioning from a local test server to a live server extremely easy as well.

Features

XAMPP is regularly updated to the latest releases of Apache, MariaDB, PHP and Perl. It also comes with a number of other modules including OpenSSL, phpMyAdmin, MediaWiki, Joomla, WordPress and more. Self-contained, multiple instances of XAMPP can exist on a single computer, and any given instance can be copied from one computer to another. XAMPP is offered in both a full and a standard version (Smaller version).

Usage

Officially, XAMPP's designers intended it for use only as a development tool, to allow website designers and programmers to test their work on their own computers without any access to the Internet. To make this as easy as possible, many important security features are disabled by default. XAMPP has the ability to serve web pages on the World Wide Web. A special tool is provided to password-protect the most important parts of the package. XAMPP also provides support for creating and manipulating databases in MariaDB and SQLite among others. Once XAMPP is installed, it is possible to treat a localhost like a remote host by connecting using an FTP client. Using a program like FileZilla has many advantages when installing a content management system (CMS) like Joomla or WordPress. It is also possible to connect to localhost via FTP with an HTML editor.

7. Notepad++:

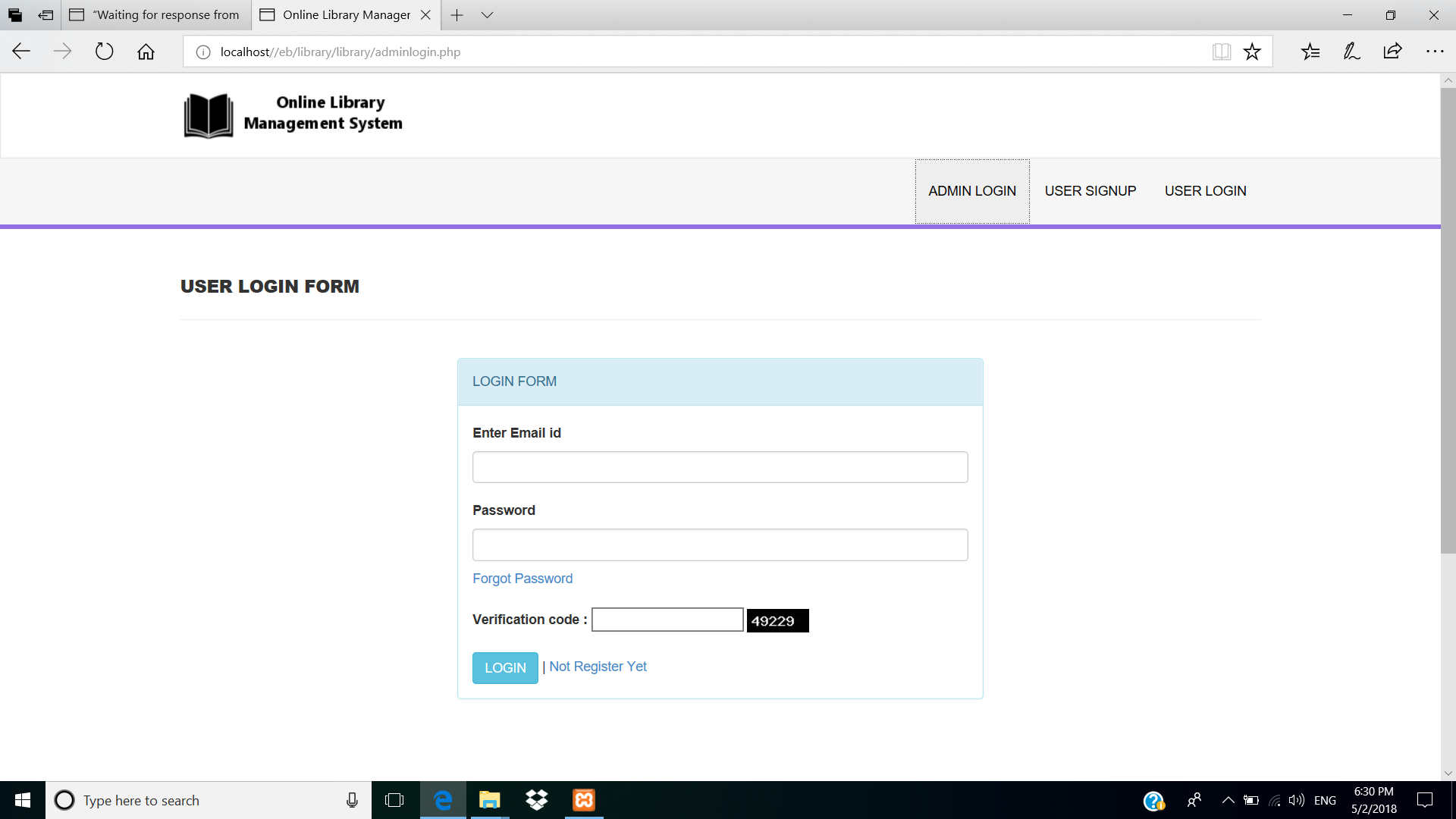
Notepad++ is a text editor and source code editor for use with Microsoft Windows. It supports tabbed editing, which allows working with multiple open files in a single window. The project's name comes from the C increment operator.

Notepad++ is distributed as free software. At first the project was hosted on SourceForge.net, from where it has been downloaded over 28 million times, and twice won the SourceForge Community Choice Award for Best Developer Tool.[4] The project was hosted on TuxFamily (fr) from 2010 to 2015; since 2015 Notepad++ has been hosted on GitHub. Notepad++ uses the Scintilla editor component.

8.4MODULES DESCRIPTION:

There are modules involving that describes the whole purpose of the project. Following are the modules.

User Module:



This module is further divided into various sub-modules describing the user in a better way:

New user register:

To sign up a new user to this system

Student Login:

So as to confirm that only an authenticated user is using the project.

Search book:

The user can search book based on book id, book name, or by author name.

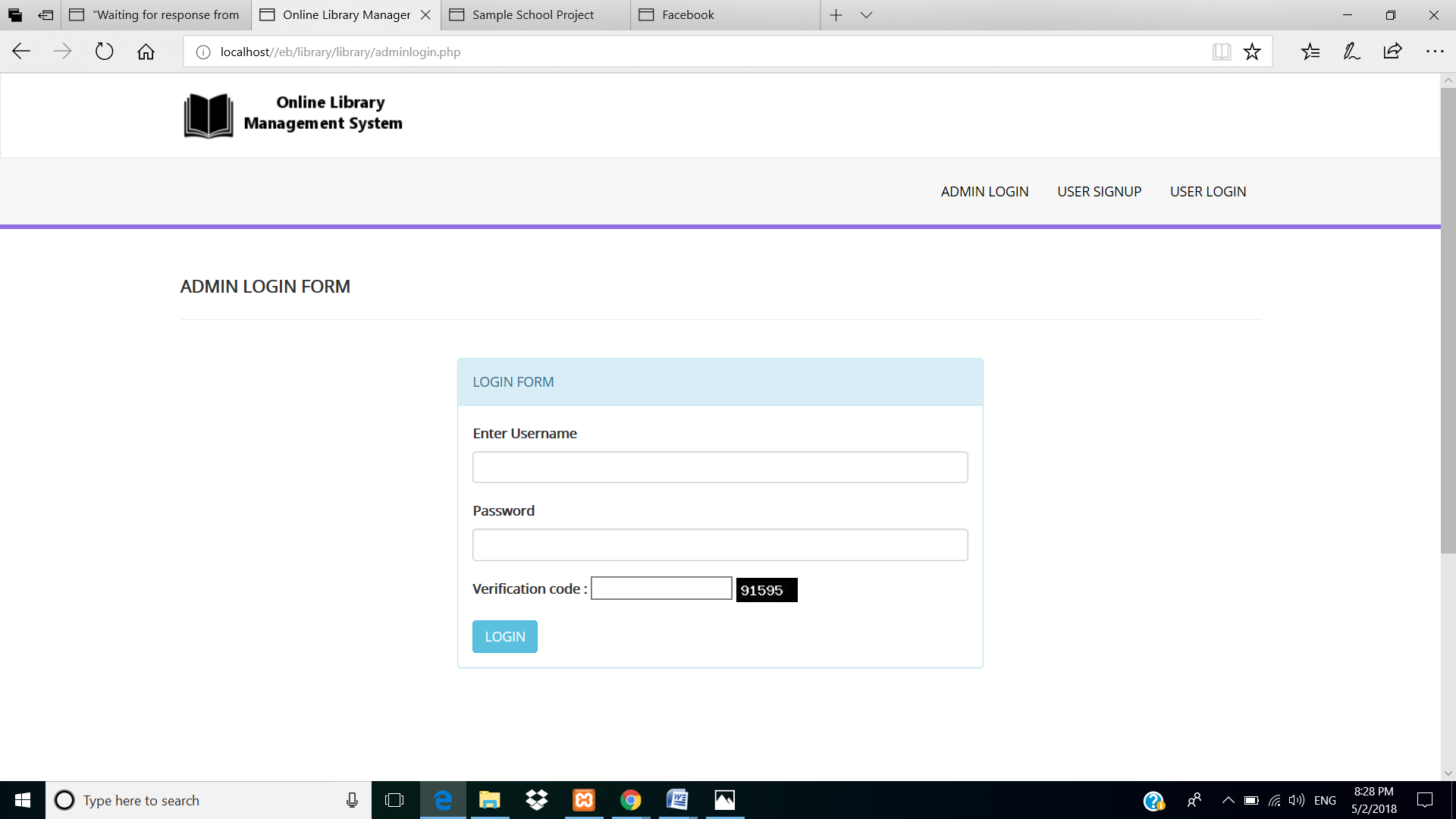
Issue Book:

To help the user get the required books issued.

Return Book:

To return the book before the last date without fine, or after the specified time duration with a late fine.

Admin Module:



It is to be operated by the admin with unique id and password. The admin is the person who decides authentication and authorization for all the different users of the application. It further can be subdivided as:

Register user

Issue Book

Maintain books in a stack, means record the availability at regular time interval.

Librarian:

Includes all the library staff who are required to enter the records in the system and keep an eye on the various activities like the issue of the book, the return of the book, non-availability of books etc. through the developed system.

### Functional Requirement of library management system

On the other side, there are those that deal with all type of technical functioning of the system.

#### Login :

#### C:\Users\Admin\Saved Games\Pictures\Screenshots\Screenshot (18).png

he is to authenticate a user, that is to know whether he or she can get access to the system. At the time of login, the user will be required to enter their user id and password.

If for any user these fields don’t match, then the user will not be allowed to use the system. For this, the user id is stored at the time of registration.

This library managememt system must only allow a user with valid id and password to become the beneficiary.After this authorization takes place, to know what all are the levels a particular user can access to.

Also after finishing the work user must log out of the system to prevent the transaction from any intruder.

#### User Sign Up:

#### C:\Users\Admin\Saved Games\Pictures\Screenshots\Screenshot (17).png

This is performed when the user is new to the system and wants to become a part, that wants to create a new account. For this system must verify all the user’s information. And also the information must be removed if found invalid or irrelevant.

#### Adding New book in the library:

#### C:\Users\Admin\Saved Games\Pictures\Screenshots\Screenshot (34).png

This feature is used to add new books to the library by the authority. The system must enter and maintain the number of copies of each individual book.Also, the system must allocate unique id to individual books carefully.

#### Search operation:

The system must provide the facility of searching books based on their unique identity, the name of the book, author name. There must be some filters available to search with keywords. Some table view of the searches must be available.

#### Issue and Return book:

#### C:\Users\Admin\Saved Games\Pictures\Screenshots\Screenshot (35).png

This is for issuing and returning books and also maintaining the issue and return status in the database timely. The system must be performing well with storing the issue data into the database. The update number of books feature must be working fine.

Before issuing any book, the system must firstly check for its availability, if it is available in stock or not. Along with issuing the books, the return dates must be shared with the students and must be entered into the database also.

#### Fine charge:

In case of failure in return on or before the last date, the student will be charged penalty fine, based on the per day rate multiplied by the number of days. The system must update this status time to time.

Also once the fine has been paid, the status must be updated again shortly.

**8.5 SAMPLE CODE:**

Admin login:

<?php

session\_start();

error\_reporting(0);

include('includes/config.php');

if($\_SESSION['alogin']!=''){

$\_SESSION['alogin']='';

}

if(isset($\_POST['login']))

{

//code for captach verification

if ($\_POST["vercode"] != $\_SESSION["vercode"] OR $\_SESSION["vercode"]=='') {

echo "<script>alert('Incorrect verification code');</script>" ;

}

else {

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

$password=md5($\_POST['password']);

$sql ="SELECT UserName,Password FROM admin WHERE UserName=:username and Password=:password";

$query= $dbh -> prepare($sql);

$query-> bindParam(':username', $username, PDO::PARAM\_STR);

$query-> bindParam(':password', $password, PDO::PARAM\_STR);

$query-> execute();

$results=$query->fetchAll(PDO::FETCH\_OBJ);

if($query->rowCount() > 0)

{

$\_SESSION['alogin']=$\_POST['username'];

echo "<script type='text/javascript'> document.location ='admin/dashboard.php'; </script>";

} else{

echo "<script>alert('Invalid Details');</script>";

}

}

}

?>

<!DOCTYPE html>

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

<head>

<meta charset="utf-8" />

<meta name="viewport" content="width=device-width, initial-scale=1, maximum-scale=1" />

<meta name="description" content="" />

<meta name="author" content="" />

<title>Online Library Management System</title>

<!-- BOOTSTRAP CORE STYLE -->

<link href="assets/css/bootstrap.css" rel="stylesheet" />

<!-- FONT AWESOME STYLE -->

<link href="assets/css/font-awesome.css" rel="stylesheet" />

<!-- CUSTOM STYLE -->

<link href="assets/css/style.css" rel="stylesheet" />

<!-- GOOGLE FONT -->

<link href='http://fonts.googleapis.com/css?family=Open+Sans' rel='stylesheet' type='text/css' />

</head>

<body>

<!------MENU SECTION START-->

<?php include('includes/header.php');?>

<!-- MENU SECTION END-->

<div class="content-wrapper">

<div class="container">

<div class="row pad-botm">

<div class="col-md-12">

<h4 class="header-line">ADMIN LOGIN FORM</h4>

</div>

</div>

<!--LOGIN PANEL START-->

<div class="row">

<div class="col-md-6 col-sm-6 col-xs-12 col-md-offset-3" >

<div class="panel panel-info">

<div class="panel-heading">

LOGIN FORM

</div>

<div class="panel-body">

<form role="form" method="post">

<div class="form-group">

<label>Enter Username</label>

<input class="form-control" type="text" name="username" autocomplete="off" required />

</div>

<div class="form-group">

<label>Password</label>

<input class="form-control" type="password" name="password" autocomplete="off" required />

</div>

<div class="form-group">

<label>Verification code : </label>

<input type="text" name="vercode" maxlength="5" autocomplete="off" required style="width: 150px; height: 25px;" />&nbsp;<img src="captcha.php">

</div>

<button type="submit" name="login" class="btn btn-info">LOGIN </button>

</form>

</div>

</div>

</div>

</div>

<!---LOGIN PABNEL END-->

</div>

</div>

<!-- CONTENT-WRAPPER SECTION END-->

<?php include('includes/footer.php');?>

<!-- FOOTER SECTION END-->

<script src="assets/js/jquery-1.10.2.js"></script>

<!-- BOOTSTRAP SCRIPTS -->

<script src="assets/js/bootstrap.js"></script>

<!-- CUSTOM SCRIPTS -->

<script src="assets/js/custom.js"></script>

</script>

</body>

</html>

Change password:

<?php

session\_start();

include('includes/config.php');

error\_reporting(0);

if(strlen($\_SESSION['login'])==0)

{

header('location:index.php');

}

else{

if(isset($\_POST['change']))

{

$password=md5($\_POST['password']);

$newpassword=md5($\_POST['newpassword']);

$email=$\_SESSION['login'];

$sql ="SELECT Password FROM tblstudents WHERE EmailId=:email and Password=:password";

$query= $dbh -> prepare($sql);

$query-> bindParam(':email', $email, PDO::PARAM\_STR);

$query-> bindParam(':password', $password, PDO::PARAM\_STR);

$query-> execute();

$results = $query -> fetchAll(PDO::FETCH\_OBJ);

if($query -> rowCount() > 0)

{

$con="update tblstudents set Password=:newpassword where EmailId=:email";

$chngpwd1 = $dbh->prepare($con);

$chngpwd1-> bindParam(':email', $email, PDO::PARAM\_STR);

$chngpwd1-> bindParam(':newpassword', $newpassword, PDO::PARAM\_STR);

$chngpwd1->execute();

$msg="Your Password succesfully changed";

}

else {

$error="Your current password is wrong";

}

}

?>

<!DOCTYPE html>

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

<head>

<meta charset="utf-8" />

<meta name="viewport" content="width=device-width, initial-scale=1, maximum-scale=1" />

<meta name="description" content="" />

<meta name="author" content="" />

<title>Online Library Management System | </title>

<!-- BOOTSTRAP CORE STYLE -->

<link href="assets/css/bootstrap.css" rel="stylesheet" />

<!-- FONT AWESOME STYLE -->

<link href="assets/css/font-awesome.css" rel="stylesheet" />

<!-- CUSTOM STYLE -->

<link href="assets/css/style.css" rel="stylesheet" />

<!-- GOOGLE FONT -->

<link href='http://fonts.googleapis.com/css?family=Open+Sans' rel='stylesheet' type='text/css' />

<style>

.errorWrap {

padding: 10px;

margin: 0 0 20px 0;

background: #fff;

border-left: 4px solid #dd3d36;

-webkit-box-shadow: 0 1px 1px 0 rgba(0,0,0,.1);

box-shadow: 0 1px 1px 0 rgba(0,0,0,.1);

}

.succWrap{

padding: 10px;

margin: 0 0 20px 0;

background: #fff;

border-left: 4px solid #5cb85c;

-webkit-box-shadow: 0 1px 1px 0 rgba(0,0,0,.1);

box-shadow: 0 1px 1px 0 rgba(0,0,0,.1);

}

</style>

</head>

<script type="text/javascript">

function valid()

{

if(document.chngpwd.newpassword.value!= document.chngpwd.confirmpassword.value)

{

alert("New Password and Confirm Password Field do not match !!");

document.chngpwd.confirmpassword.focus();

return false;

}

return true;

}

</script>

<body>

<!------MENU SECTION START-->

<?php include('includes/header.php');?>

<!-- MENU SECTION END-->

<div class="content-wrapper">

<div class="container">

<div class="row pad-botm">

<div class="col-md-12">

<h4 class="header-line">User Change Password</h4>

</div>

</div>

<?php if($error){?><div class="errorWrap"><strong>ERROR</strong>:<?php echo htmlentities($error); ?> </div><?php }

else if($msg){?><div class="succWrap"><strong>SUCCESS</strong>:<?php echo htmlentities($msg); ?> </div><?php }?>

<!--LOGIN PANEL START-->

<div class="row">

<div class="col-md-6 col-sm-6 col-xs-12 col-md-offset-3" >

<div class="panel panel-info">

<div class="panel-heading">

Change Password

</div>

<div class="panel-body">

<form role="form" method="post" onSubmit="return valid();" name="chngpwd">

<div class="form-group">

<label>Current Password</label>

<input class="form-control" type="password" name="password" autocomplete="off" required />

</div>

<div class="form-group">

<label>Enter Password</label>

<input class="form-control" type="password" name="newpassword" autocomplete="off" required />

</div>

<div class="form-group">

<label>Confirm Password </label>

<input class="form-control" type="password" name="confirmpassword" autocomplete="off" required />

</div>

<button type="submit" name="change" class="btn btn-info">Chnage </button>

</form>

</div>

</div>

</div>

</div>

<!---LOGIN PABNEL END-->

</div>

</div>

<!-- CONTENT-WRAPPER SECTION END-->

<?php include('includes/footer.php');?>

<!-- FOOTER SECTION END-->

<script src="assets/js/jquery-1.10.2.js"></script>

<!-- BOOTSTRAP SCRIPTS -->

<script src="assets/js/bootstrap.js"></script>

<!-- CUSTOM SCRIPTS -->

<script src="assets/js/custom.js"></script>

</body>

</html>

<?php } ?>

Check availability:

<?php

require\_once("includes/config.php");

// code user email availablity

if(!empty($\_POST["emailid"])) {

$email= $\_POST["emailid"];

if (filter\_var($email, FILTER\_VALIDATE\_EMAIL)===false) {

echo "error : You did not enter a valid email.";

}

else {

$sql ="SELECT EmailId FROM tblstudents WHERE EmailId=:email";

$query= $dbh -> prepare($sql);

$query-> bindParam(':email', $email, PDO::PARAM\_STR);

$query-> execute();

$results = $query -> fetchAll(PDO::FETCH\_OBJ);

$cnt=1;

if($query -> rowCount() > 0)

{

echo "<span style='color:red'> Email already exists .</span>";

echo "<script>$('#submit').prop('disabled',true);</script>";

} else{

echo "<span style='color:green'> Email available for Registration .</span>";

echo "<script>$('#submit').prop('disabled',false);</script>";

}

}

}

?>

Check availability:

<?php

require\_once("includes/config.php");

// code user email availablity

if(!empty($\_POST["emailid"])) {

$email= $\_POST["emailid"];

if (filter\_var($email, FILTER\_VALIDATE\_EMAIL)===false) {

echo "error : You did not enter a valid email.";

}

else {

$sql ="SELECT EmailId FROM tblstudents WHERE EmailId=:email";

$query= $dbh -> prepare($sql);

$query-> bindParam(':email', $email, PDO::PARAM\_STR);

$query-> execute();

$results = $query -> fetchAll(PDO::FETCH\_OBJ);

$cnt=1;

if($query -> rowCount() > 0)

{

echo "<span style='color:red'> Email already exists .</span>";

echo "<script>$('#submit').prop('disabled',true);</script>";

} else{

echo "<span style='color:green'> Email available for Registration .</span>";

echo "<script>$('#submit').prop('disabled',false);</script>";

}

}

}

?>

Index:

<?php

session\_start();

error\_reporting(0);

include('includes/config.php');

if($\_SESSION['login']!=''){

$\_SESSION['login']='';

}

if(isset($\_POST['login']))

{

//code for captach verification

if ($\_POST["vercode"] != $\_SESSION["vercode"] OR $\_SESSION["vercode"]=='') {

echo "<script>alert('Incorrect verification code');</script>" ;

}

else {

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

$password=md5($\_POST['password']);

$sql ="SELECT EmailId,Password,StudentId,Status FROM tblstudents WHERE EmailId=:email and Password=:password";

$query= $dbh -> prepare($sql);

$query-> bindParam(':email', $email, PDO::PARAM\_STR);

$query-> bindParam(':password', $password, PDO::PARAM\_STR);

$query-> execute();

$results=$query->fetchAll(PDO::FETCH\_OBJ);

if($query->rowCount() > 0)

{

foreach ($results as $result) {

$\_SESSION['stdid']=$result->StudentId;

if($result->Status==1)

{

$\_SESSION['login']=$\_POST['emailid'];

echo "<script type='text/javascript'> document.location ='dashboard.php'; </script>";

} else {

echo "<script>alert('Your Account Has been blocked .Please contact admin');</script>";

}

}

}

else{

echo "<script>alert('Invalid Details');</script>";

}

}

}

?>

<!DOCTYPE html>

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

<head>

<meta charset="utf-8" />

<meta name="viewport" content="width=device-width, initial-scale=1, maximum-scale=1" />

<meta name="description" content="" />

<meta name="author" content="" />

<title>Online Library Management System | </title>

<!-- BOOTSTRAP CORE STYLE -->

<link href="assets/css/bootstrap.css" rel="stylesheet" />

<!-- FONT AWESOME STYLE -->

<link href="assets/css/font-awesome.css" rel="stylesheet" />

<!-- CUSTOM STYLE -->

<link href="assets/css/style.css" rel="stylesheet" />

<!-- GOOGLE FONT -->

<link href='http://fonts.googleapis.com/css?family=Open+Sans' rel='stylesheet' type='text/css' />

</head>

<body>

<!------MENU SECTION START-->

<?php include('includes/header.php');?>

<!-- MENU SECTION END-->

<div class="content-wrapper">

<div class="container">

<div class="row pad-botm">

<div class="col-md-12">

<h4 class="header-line">USER LOGIN FORM</h4>

</div>

</div>

<!--LOGIN PANEL START-->

<div class="row">

<div class="col-md-6 col-sm-6 col-xs-12 col-md-offset-3" >

<div class="panel panel-info">

<div class="panel-heading">

LOGIN FORM

</div>

<div class="panel-body">

<form role="form" method="post">

<div class="form-group">

<label>Enter Email id</label>

<input class="form-control" type="text" name="emailid" required autocomplete="off" />

</div>

<div class="form-group">

<label>Password</label>

<input class="form-control" type="password" name="password" required autocomplete="off" />

<p class="help-block"><a href="user-forgot-password.php">Forgot Password</a></p>

</div>

<div class="form-group">

<label>Verification code : </label>

<input type="text" class="form-control1" name="vercode" maxlength="5" autocomplete="off" required style="height:25px;" />&nbsp;<img src="captcha.php">

</div>

<button type="submit" name="login" class="btn btn-info">LOGIN </button> | <a href="signup.php">Not Register Yet</a>

</form>

</div>

</div>

</div>

</div>

<!---LOGIN PABNEL END-->

</div>

</div>

<!-- CONTENT-WRAPPER SECTION END-->

<?php include('includes/footer.php');?>

<!-- FOOTER SECTION END-->

<script src="assets/js/jquery-1.10.2.js"></script>

<!-- BOOTSTRAP SCRIPTS -->

<script src="assets/js/bootstrap.js"></script>

<!-- CUSTOM SCRIPTS -->

<script src="assets/js/custom.js"></script>

</body>

</html>

Issued books:

<?php

session\_start();

error\_reporting(0);

include('includes/config.php');

if(strlen($\_SESSION['login'])==0)

{

header('location:index.php');

}

else{

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

{

$id=$\_GET['del'];

$sql = "delete from tblbooks WHERE id=:id";

$query = $dbh->prepare($sql);

$query -> bindParam(':id',$id, PDO::PARAM\_STR);

$query -> execute();

$\_SESSION['delmsg']="Category deleted scuccessfully ";

header('location:manage-books.php');

}

?>

<!DOCTYPE html>

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

<head>

<meta charset="utf-8" />

<meta name="viewport" content="width=device-width, initial-scale=1, maximum-scale=1" />

<meta name="description" content="" />

<meta name="author" content="" />

<title>Online Library Management System | Manage Issued Books</title>

<!-- BOOTSTRAP CORE STYLE -->

<link href="assets/css/bootstrap.css" rel="stylesheet" />

<!-- FONT AWESOME STYLE -->

<link href="assets/css/font-awesome.css" rel="stylesheet" />

<!-- DATATABLE STYLE -->

<link href="assets/js/dataTables/dataTables.bootstrap.css" rel="stylesheet" />

<!-- CUSTOM STYLE -->

<link href="assets/css/style.css" rel="stylesheet" />

<!-- GOOGLE FONT -->

<link href='http://fonts.googleapis.com/css?family=Open+Sans' rel='stylesheet' type='text/css' />

</head>

<body>

<!------MENU SECTION START-->

<?php include('includes/header.php');?>

<!-- MENU SECTION END-->

<div class="content-wrapper">

<div class="container">

<div class="row pad-botm">

<div class="col-md-12">

<h4 class="header-line">Manage Issued Books</h4>

</div>

<div class="row">

<div class="col-md-12">

<!-- Advanced Tables -->

<div class="panel panel-default">

<div class="panel-heading">

Issued Books

</div>

<div class="panel-body">

<div class="table-responsive">

<table class="table table-striped table-bordered table-hover" id="dataTables-example">

<thead>

<tr>

<th>#</th>

<th>Book Name</th>

<th>ISBN </th>

<th>Issued Date</th>

<th>Return Date</th>

<th>Fine in(USD)</th>

</tr>

</thead>

<tbody>

<?php

$sid=$\_SESSION['stdid'];

$sql="SELECT tblbooks.BookName,tblbooks.ISBNNumber,tblissuedbookdetails.IssuesDate,tblissuedbookdetails.ReturnDate,tblissuedbookdetails.id as rid,tblissuedbookdetails.fine from tblissuedbookdetails join tblstudents on tblstudents.StudentId=tblissuedbookdetails.StudentId join tblbooks on tblbooks.id=tblissuedbookdetails.BookId where tblstudents.StudentId=:sid order by tblissuedbookdetails.id desc";

$query = $dbh -> prepare($sql);

$query-> bindParam(':sid', $sid, PDO::PARAM\_STR);

$query->execute();

$results=$query->fetchAll(PDO::FETCH\_OBJ);

$cnt=1;

if($query->rowCount() > 0)

{

foreach($results as $result)

{ ?>

<tr class="odd gradeX">

<td class="center"><?php echo htmlentities($cnt);?></td>

<td class="center"><?php echo htmlentities($result->BookName);?></td>

<td class="center"><?php echo htmlentities($result->ISBNNumber);?></td>

<td class="center"><?php echo htmlentities($result->IssuesDate);?></td>

<td class="center"><?php if($result->ReturnDate=="")

{?>

<span style="color:red">

<?php echo htmlentities("Not Return Yet"); ?>

</span>

<?php } else {

echo htmlentities($result->ReturnDate);

}

?></td>

<td class="center"><?php echo htmlentities($result->fine);?></td>

</tr>

<?php $cnt=$cnt+1;}} ?>

</tbody>

</table>

</div>

</div>

</div>

<!--End Advanced Tables -->

</div>

</div>

</div>

</div>

</div>

<!-- CONTENT-WRAPPER SECTION END-->

<?php include('includes/footer.php');?>

<!-- FOOTER SECTION END-->

<!-- JAVASCRIPT FILES PLACED AT THE BOTTOM TO REDUCE THE LOADING TIME -->

<!-- CORE JQUERY -->

<script src="assets/js/jquery-1.10.2.js"></script>

<!-- BOOTSTRAP SCRIPTS -->

<script src="assets/js/bootstrap.js"></script>

<!-- DATATABLE SCRIPTS -->

<script src="assets/js/dataTables/jquery.dataTables.js"></script>

<script src="assets/js/dataTables/dataTables.bootstrap.js"></script>

<!-- CUSTOM SCRIPTS -->

<script src="assets/js/custom.js"></script>

</body>

</html>

<?php } ?>

Logout:

<?php

session\_start();

$\_SESSION = array();

if (ini\_get("session.use\_cookies")) {

$params = session\_get\_cookie\_params();

setcookie(session\_name(), '', time() - 60\*60,

$params["path"], $params["domain"],

$params["secure"], $params["httponly"]

);

}

unset($\_SESSION['login']);

session\_destroy(); // destroy session

header("location:index.php");

?>

Logout

<?php

session\_start();

$\_SESSION = array();

if (ini\_get("session.use\_cookies")) {

$params = session\_get\_cookie\_params();

setcookie(session\_name(), '', time() - 60\*60,

$params["path"], $params["domain"],

$params["secure"], $params["httponly"]

);

}

unset($\_SESSION['login']);

session\_destroy(); // destroy session

header("location:index.php");

?>

Forgot password

<?php

session\_start();

error\_reporting(0);

include('includes/config.php');

if(isset($\_POST['change']))

{

//code for captach verification

if ($\_POST["vercode"] != $\_SESSION["vercode"] OR $\_SESSION["vercode"]=='') {

echo "<script>alert('Incorrect verification code');</script>" ;

}

else {

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

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

$newpassword=md5($\_POST['newpassword']);

$sql ="SELECT EmailId FROM tblstudents WHERE EmailId=:email and MobileNumber=:mobile";

$query= $dbh -> prepare($sql);

$query-> bindParam(':email', $email, PDO::PARAM\_STR);

$query-> bindParam(':mobile', $mobile, PDO::PARAM\_STR);

$query-> execute();

$results = $query -> fetchAll(PDO::FETCH\_OBJ);

if($query -> rowCount() > 0)

{

$con="update tblstudents set Password=:newpassword where EmailId=:email and MobileNumber=:mobile";

$chngpwd1 = $dbh->prepare($con);

$chngpwd1-> bindParam(':email', $email, PDO::PARAM\_STR);

$chngpwd1-> bindParam(':mobile', $mobile, PDO::PARAM\_STR);

$chngpwd1-> bindParam(':newpassword', $newpassword, PDO::PARAM\_STR);

$chngpwd1->execute();

echo "<script>alert('Your Password succesfully changed');</script>";

}

else {

echo "<script>alert('Email id or Mobile no is invalid');</script>";

}

}

}

?>

<!DOCTYPE html>

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

<head>

<meta charset="utf-8" />

<meta name="viewport" content="width=device-width, initial-scale=1, maximum-scale=1" />

<meta name="description" content="" />

<meta name="author" content="" />

<title>Online Library Management System | Password Recovery </title>

<!-- BOOTSTRAP CORE STYLE -->

<link href="assets/css/bootstrap.css" rel="stylesheet" />

<!-- FONT AWESOME STYLE -->

<link href="assets/css/font-awesome.css" rel="stylesheet" />

<!-- CUSTOM STYLE -->

<link href="assets/css/style.css" rel="stylesheet" />

<!-- GOOGLE FONT -->

<link href='http://fonts.googleapis.com/css?family=Open+Sans' rel='stylesheet' type='text/css' />

<script type="text/javascript">

function valid()

{

if(document.chngpwd.newpassword.value!= document.chngpwd.confirmpassword.value)

{

alert("New Password and Confirm Password Field do not match !!");

document.chngpwd.confirmpassword.focus();

return false;

}

return true;

}

</script>

</head>

<body>

<!------MENU SECTION START-->

<?php include('includes/header.php');?>

<!-- MENU SECTION END-->

<div class="content-wrapper">

<div class="container">

<div class="row pad-botm">

<div class="col-md-12">

<h4 class="header-line">User Password Recovery</h4>

</div>

</div>

<!--LOGIN PANEL START-->

<div class="row">

<div class="col-md-6 col-sm-6 col-xs-12 col-md-offset-3" >

<div class="panel panel-info">

<div class="panel-heading">

LOGIN FORM

</div>

<div class="panel-body">

<form role="form" name="chngpwd" method="post" onSubmit="return valid();">

<div class="form-group">

<label>Enter Reg Email id</label>

<input class="form-control" type="email" name="email" required autocomplete="off" />

</div>

<div class="form-group">

<label>Enter Reg Mobile No</label>

<input class="form-control" type="text" name="mobile" required autocomplete="off" />

</div>

<div class="form-group">

<label>Password</label>

<input class="form-control" type="password" name="newpassword" required autocomplete="off" />

</div>

<div class="form-group">

<label>ConfirmPassword</label>

<input class="form-control" type="password" name="confirmpassword" required autocomplete="off" />

</div>

<div class="form-group">

<label>Verification code : </label>

<input type="text" class="form-control1" name="vercode" maxlength="5" autocomplete="off" required style="height:25px;" />&nbsp;<img src="captcha.php">

</div>

<button type="submit" name="change" class="btn btn-info">Chnage Password</button> | <a href="index.php">Login</a>

</form>

</div>

</div>

</div>

</div>

<!---LOGIN PABNEL END-->

</div>

</div>

<!-- CONTENT-WRAPPER SECTION END-->

<?php include('includes/footer.php');?>

<!-- FOOTER SECTION END-->

<script src="assets/js/jquery-1.10.2.js"></script>

<!-- BOOTSTRAP SCRIPTS -->

<script src="assets/js/bootstrap.js"></script>

<!-- CUSTOM SCRIPTS -->

<script src="assets/js/custom.js"></script>

</body>

</html>

Dashboard:

<?php

session\_start();

error\_reporting(0);

include('includes/config.php');

if(strlen($\_SESSION['login'])==0)

{

header('location:index.php');

}

else{?>

<!DOCTYPE html>

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

<head>

<meta charset="utf-8" />

<meta name="viewport" content="width=device-width, initial-scale=1, maximum-scale=1" />

<meta name="description" content="" />

<meta name="author" content="" />

<title>Online Library Management System | User Dash Board</title>

<!-- BOOTSTRAP CORE STYLE -->

<link href="assets/css/bootstrap.css" rel="stylesheet" />

<!-- FONT AWESOME STYLE -->

<link href="assets/css/font-awesome.css" rel="stylesheet" />

<!-- CUSTOM STYLE -->

<link href="assets/css/style.css" rel="stylesheet" />

<!-- GOOGLE FONT -->

<link href='http://fonts.googleapis.com/css?family=Open+Sans' rel='stylesheet' type='text/css' />

</head>

<body>

<!------MENU SECTION START-->

<?php include('includes/header.php');?>

<!-- MENU SECTION END-->

<div class="content-wrapper">

<div class="container">

<div class="row pad-botm">

<div class="col-md-12">

<h4 class="header-line">ADMIN DASHBOARD</h4>

</div>

</div>

<div class="row">

<div class="col-md-3 col-sm-3 col-xs-6">

<div class="alert alert-info back-widget-set text-center">

<i class="fa fa-bars fa-5x"></i>

<?php

$sid=$\_SESSION['stdid'];

$sql1 ="SELECT id from tblissuedbookdetails where StudentID=:sid";

$query1 = $dbh -> prepare($sql1);

$query1->bindParam(':sid',$sid,PDO::PARAM\_STR);

$query1->execute();

$results1=$query1->fetchAll(PDO::FETCH\_OBJ);

$issuedbooks=$query1->rowCount();

?>

<h3><?php echo htmlentities($issuedbooks);?> </h3>

Book Issued

</div>

</div>

<div class="col-md-3 col-sm-3 col-xs-6">

<div class="alert alert-warning back-widget-set text-center">

<i class="fa fa-recycle fa-5x"></i>

<?php

$rsts=0;

$sql2 ="SELECT id from tblissuedbookdetails where StudentID=:sid and RetrunStatus=:rsts";

$query2 = $dbh -> prepare($sql2);

$query2->bindParam(':sid',$sid,PDO::PARAM\_STR);

$query2->bindParam(':rsts',$rsts,PDO::PARAM\_STR);

$query2->execute();

$results2=$query2->fetchAll(PDO::FETCH\_OBJ);

$returnedbooks=$query2->rowCount();

?>

<h3><?php echo htmlentities($returnedbooks);?></h3>

Books Not Returned Yet

</div>

</div>

</div>

</div>

</div>

<!-- CONTENT-WRAPPER SECTION END-->

<?php include('includes/footer.php');?>

<!-- FOOTER SECTION END-->

<!-- JAVASCRIPT FILES PLACED AT THE BOTTOM TO REDUCE THE LOADING TIME -->

<!-- CORE JQUERY -->

<script src="assets/js/jquery-1.10.2.js"></script>

<!-- BOOTSTRAP SCRIPTS -->

<script src="assets/js/bootstrap.js"></script>

<!-- CUSTOM SCRIPTS -->

<script src="assets/js/custom.js"></script>

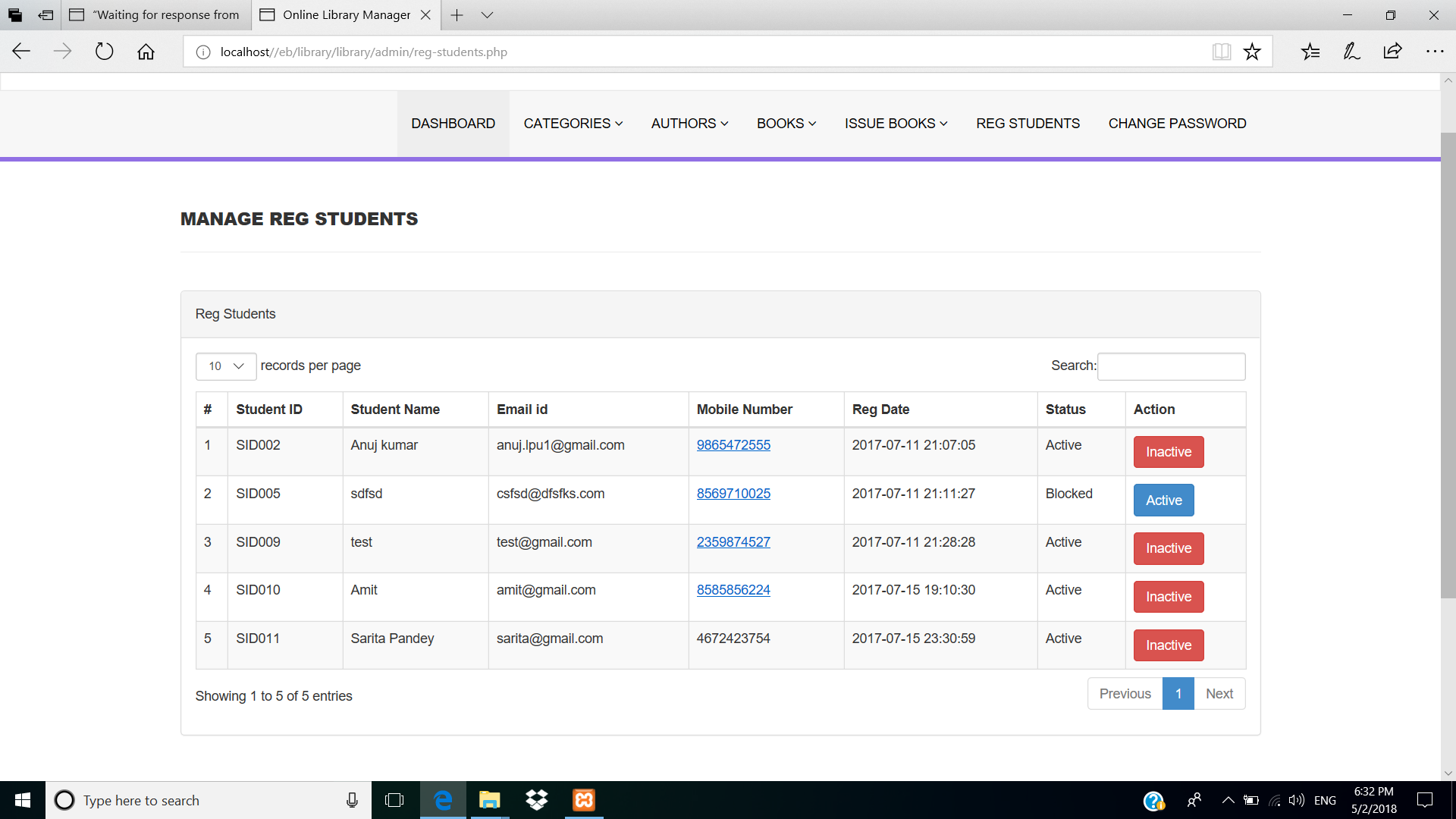
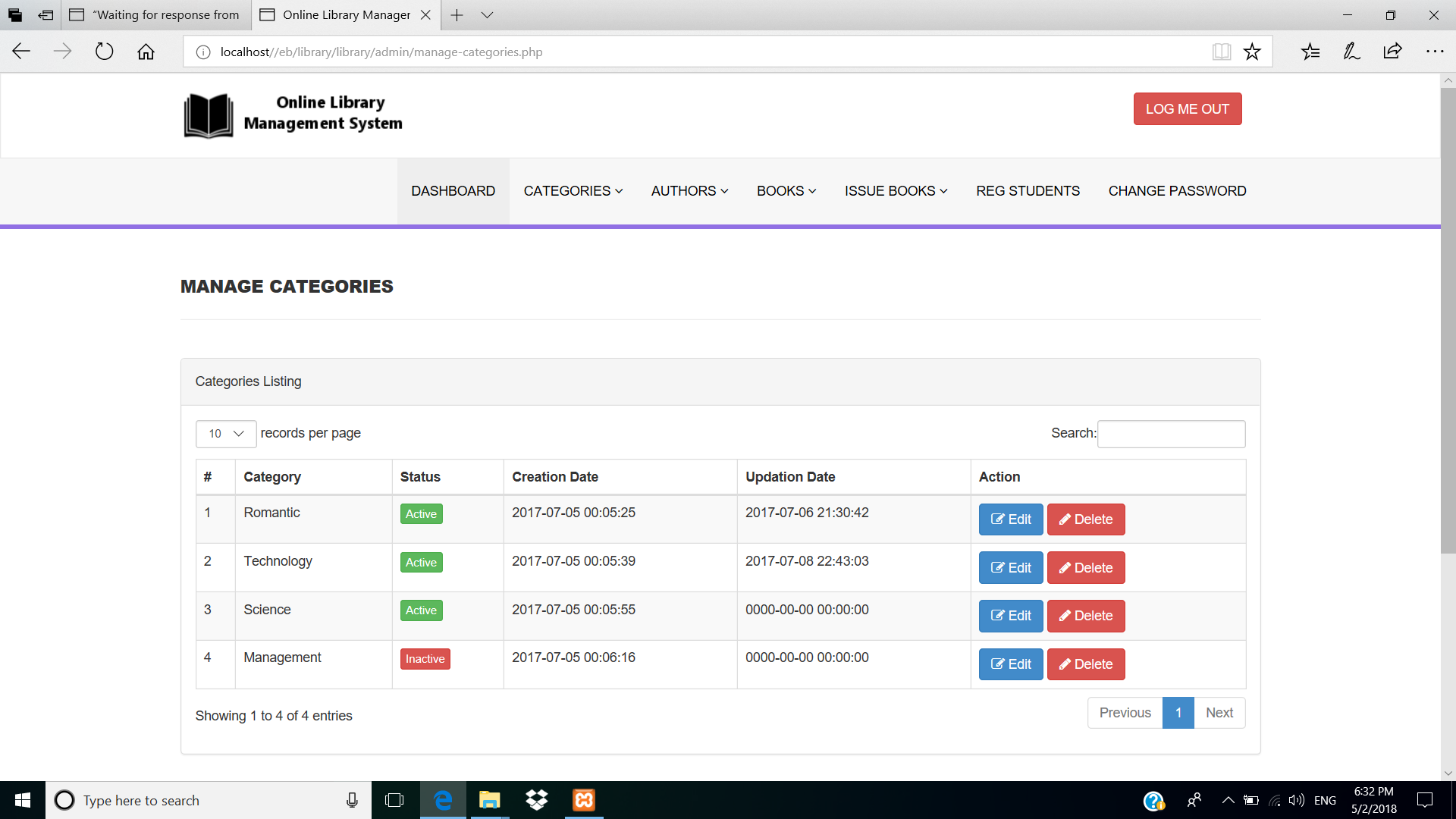
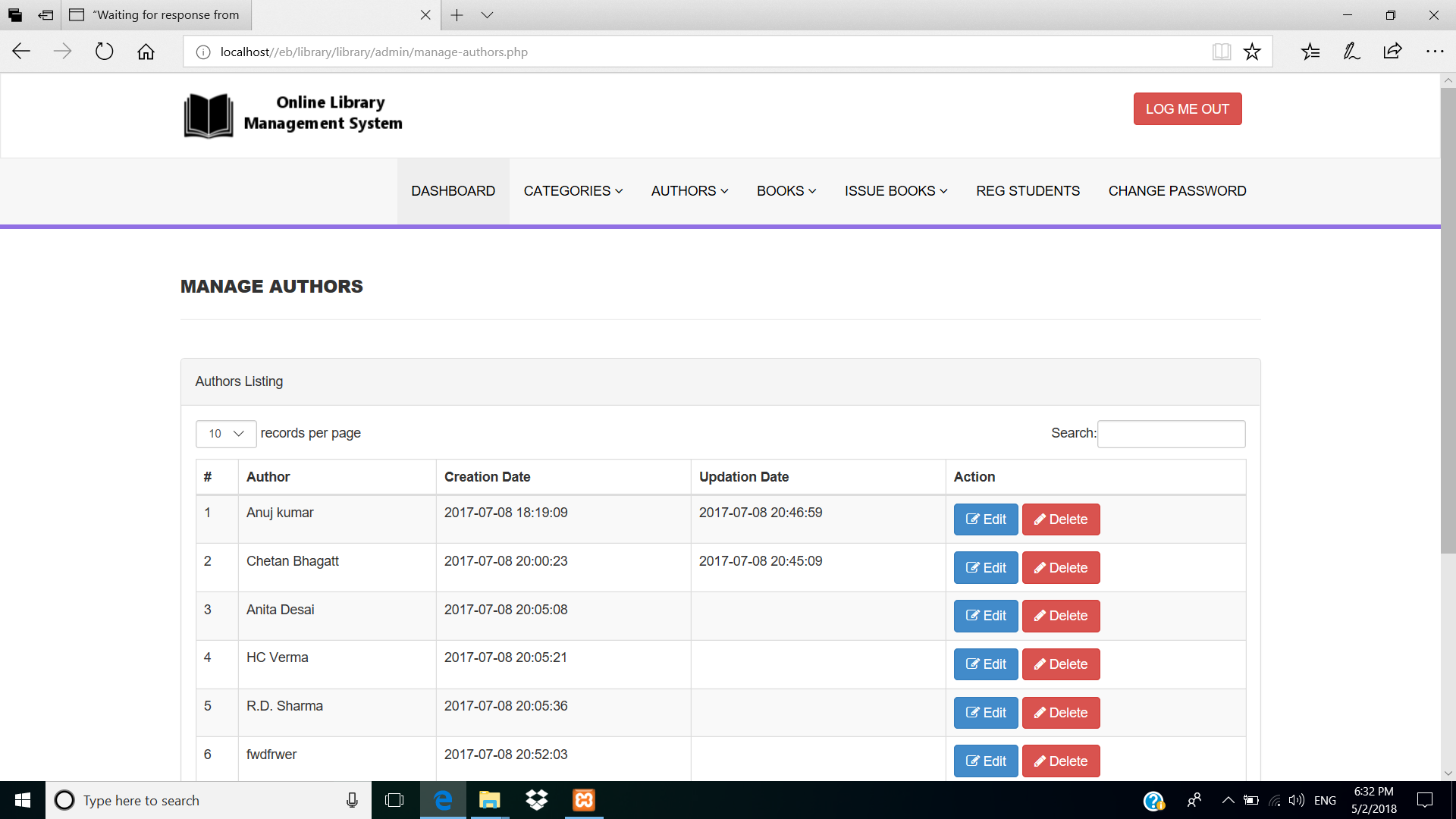
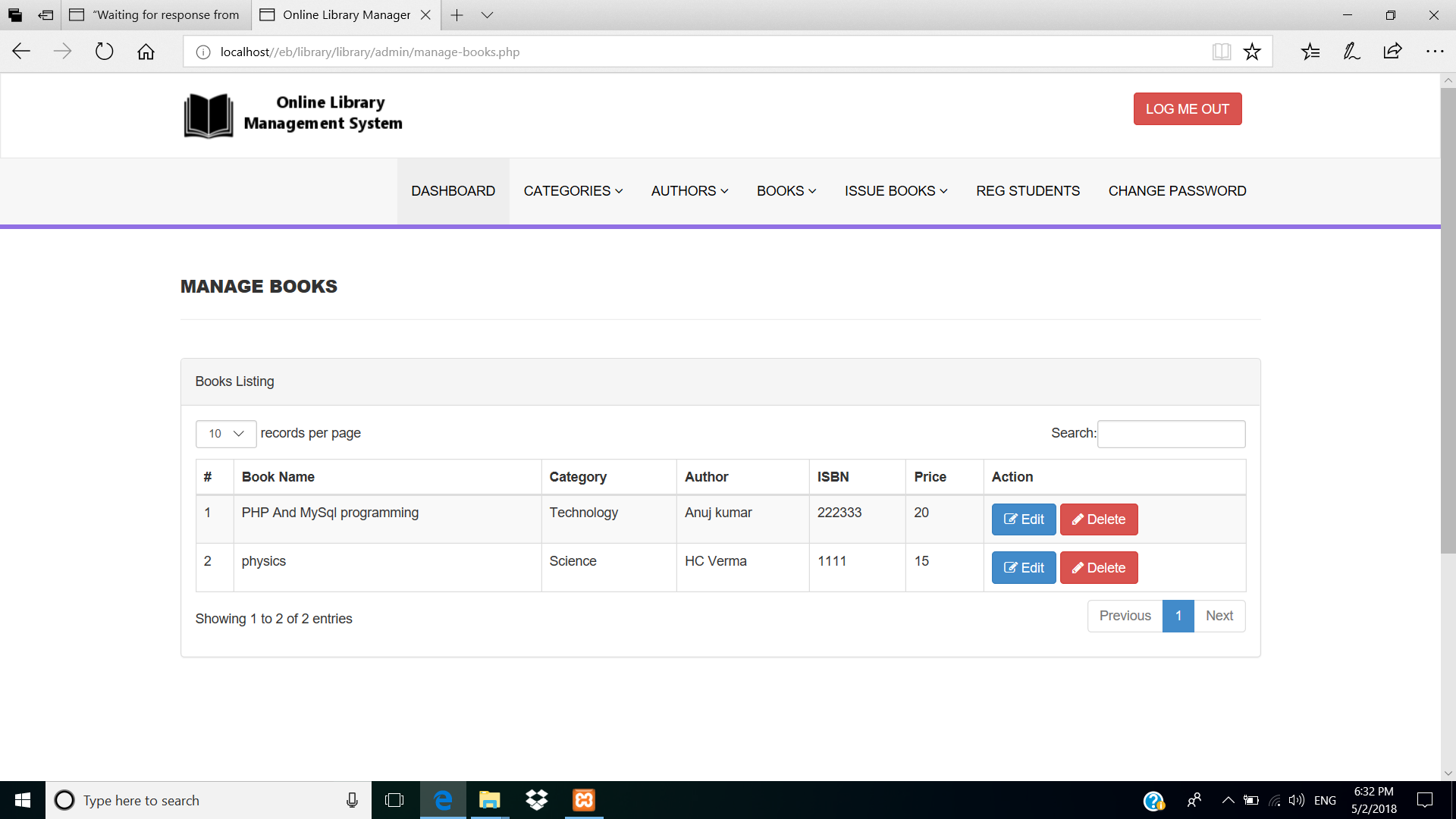
</body>

</html>

<?php } ?>

8.6 Screenshots:

Homepage:

**8.7 Conclusion:**

Thus Error generation for Authentication purpose and also for constraints to the members on the number of books allowed for a particular member.

 Member can be a Student, or Lecturer or Non teaching Staff, and respective limits for borrowing books for each type of member is set.