VISVESVARAYA TECHNOLOGICAL UNIVERSITY
BELGAUM-590014

MINI PROJECT ENTITLED

"LIBRARY MANAGEMENT WEBSITE"

Submitted in partial fulfillment of the requirements for Web Technology and its Application Laboratory.

For the academic year 2019-2020
Submitted by:

NITISH KUMAR KESHRI	**(1MV16CS063)**
HIMANSHU UPADHYAY	**(1MV16CS037)**
ASHISH KUMAR DUBEY	**(1MV16CS021)**

Mini Project carried out at
Sir M. Visvesvaraya Institute of Technology
Bangalore-562157.

Under the Guidance of
Mr. Elaiyaraja P
Asst. Professor, Dept. of CSE
Sir MVIT, Bengaluru

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
Sir M. Visvesvaraya Institute of Technology
HUNASAMARANAHALLI, Bangalore-562157.

SIR M. VISVESVARAYA INSTITUTE OF TECHNOLOGY BANGALORE-562157.

DEPARTMENT OF COMPUTER SCENCE AND ENGINEERING

CERTIFICATE

This is to certify that the Mini project report entitled **LIBRARY MANAGEMENT WEBSITE** is a bonafied work carried out by **NITISH KUMAR KESHRI (1MV16CS063), HIMANSHU UPADHYAY (1MV16CS037) and ASHISH KUMAR DUBEY (1MV16CS021)** presently VII semester student of Sir M Visvesvaraya Institute of Technology in partial fulfillment of the requirements for **Web Technology and it's Applications Laboratory** for the award of the degree of Bachelor of Engineering in Computer Science and Engineering under Visvesvaraya Technological University, Belagavi during the academic year 2019 – 2020. The Mini project report has been approved as it satisfies the academic requirements in respect of Mini project work prescribed for the Bachelor of Engineering Degree.

Signature of the Guide
Mr. Elaiyaraja P
Asst. Professor,
Dept. of CSE, Sir MVIT

Signature of the HOD
Dr.G.C.Bhanu Prakash
HOD, Dept of CSE
Sir MVIT

Internal Examiner

Name & Signature with Date

External Examiner

Name & Signature with Date

ABSTRACT

Library management system is a project which aims in developing a computerized system to maintain all the daily work of library. This project has many features which are generally not available in normal library management systems like facility of user login and a facility of teacher's login. It also has a facility of admin login through which the admin can monitor the whole system. It also has facility of an online notice board where teachers can student can put up information about workshops or seminars being held in our colleges or nearby colleges and librarian after proper verification from the concerned institution organizing the seminar can add it to the notice board. It has also a facility where student after logging in their accounts can see list of books issued and its issue date and return date and also the students can request the librarian to add new books by filling the book request form. The librarian after logging into his account i.e. admin account can generate various reports such as student report, issue report, teacher report and book report.

Overall, the project of ours is being developed to help the students as well as staff of library to maintain the library in the best way possible and reduce the human efforts.

DECLARATION

We hereby declare that the entire mini project work embodied in this dissertation has been carried out by us and no part has been submitted for any degree or diploma of any institution previously.

Place: Bengaluru Signature of Students:

Date:

NITISH KUMAR KESHRI (1MV16CS063)

HIMANSHU UPADHYAY (1MV16CS037)

ASHISH KUMAR DUBEY (1MV16CS021)

ACKNOWLEDGEMENT

It gives us immense pleasure to express our sincere gratitude to the management of **Sir M. Visvesvaraya Institute of Technology,** Bangalore for providing the opportunity and the resources to accomplish our project work in their premises.

On the path of learning, the presence of an experienced guide is indispensable and we would like to thank our guide **Mr. Elaiyaraja P, Assistant** Professor, Dept. of CSE, for her invaluable help and guidance.

We would also like to convey our regards and sincere thanks to **Prof. G.C. Bhanu Prakash,** HOD, Dept. of CSE for his suggestions, constant support and encouragement, Heartfelt and sincere thanks to **Dr. V.R.Manjunath,** Principal, Sir. MVIT for providing us with the infrastructure and facilities needed to develop our project.

We would also like to thank the staff of Department of Computer Science and Engineering and lab-in-charges for their co-operation and suggestions. Finally, we would like to thank all our friends for their help and suggestions without which completing this project would not have been possible.

-NITISH KUMAR KESHRI (1MV16CS063)
-HIMANSHU UPADHYAY (1MV16CS037)
-ASHISH KUMAR DUBEY (1MV16CS021)

Table of Contents

List of Figures

List Of Tables

Chapter 1

INTRODUCTION

1.1 Introduction to web

Web consists of billions of clients and server connected through wires and wireless networks. The web clients make requests to web server. The web server receives the request, finds the resources and returns the response to the client. When a server answers a request, it usually sends some type of content to the client. The client uses web browser to send request to the server. The server often sends response to the browser with a set of instructions written in HyperText Markup Language (HTML). All browsers know how to display HTML page to the client.

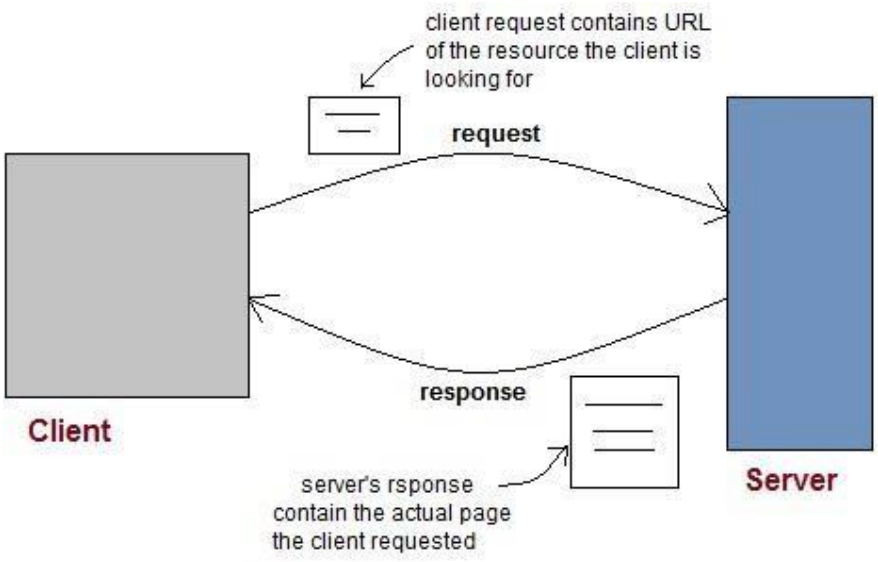


Figure 1.1 HTML—Client-Server communication

1.2 HTML

Hypertext Markup Language (HTML) is the standard markup language for creating web pages and web applications. HTML describes the structure of Web pages using markup. HTML elements are the building blocks of HTML pages.HTML elements are represented of HTML pages.

HTML elements are represented by tags.HTML tags label pieces of content such as "heading", "paragraph", "table", and so on. Browsers do not display the HTML tags, but use them to render the content of the page.

1.3 PHP

PHP is a general-purpose server-side scripting language originally designed for Web development to produce dynamic Web pages. It is one of the first developed server- side scripting languages to be embedded into an HTML source document, rather than calling an external file to process data. Ultimately, the code is interpreted by a Web server with a PHP processor module which generates the resulting Web page. It also has evolved to include a command-line interface capability and can be used in standalone graphical applications.

PHP can be deployed on most Web servers and also as a standalone shell on almost every operating system and platform free of charge. A competitor to Microsoft's Active Server Pages (ASP) server-side script engine and similar languages, PHP is installed on more than 20 million Web sites and 1 million Web servers.

In this application, PHP is used for interaction of the webpage with database. Through PHP, the user can meet the server through the HTML page. Using PHP, we can store and retrieve the information from the database using the PHP commands.

1.4 JavaScript

JavaScript (sometimes abbreviated JS) is a prototype-based scripting language that is dynamic, weakly typed, general purpose programming language and has first-class functions. It is a multi-paradigm language, supporting object-oriented, imperative, and functional programming styles.

JavaScript was formalized in the ECMA Script language standard and is primarily used in the form of client-side JavaScript, implemented as part of a Web browser in order to provide enhanced user interfaces and dynamic websites. This enables programmatic access to computational objects within a host environment. JavaScript's use in applications outside Web pages for example in PDF

documents, site-specific browsers, and desktop widgets are also significant.

In this application, JavaScript is used for validation purpose like text box validation, email validation, phone number validation. JavaScript is the good tool for validating the web- applications.

Server Side JavaScript

Meanwhile, Netscape also introduced the language for server-side scripting in Netscape Enterprise Server, first released in December, 1994.

1.5 XML

Extensible Markup Language (XML) is a markup language that defines a set of rules for encoding documents in a format that is both human-readable and machine-readable. It is defined in the XML 1.0 Specification produced by the W3C, and several other related specifications, all gratis open standards.

The design goals of XML emphasize simplicity, generality, and usability over the Internet. It is a textual data format with strong support via Unicode for the languages of the world. Although the design of XML focuses on documents, it is widely used for the representation of arbitrary data structures, for example in web services.

Many application programming interfaces (APIs) have been developed for software developers to use to process XML data, and several schema systems exist to aid in the definition of XML-based languages. As of 2009, hundreds of XML-based languages have been developed, including RSS, Atom, SOAP, and XHTML.

XML-based formats have become the default for many office-productivity tools, including Microsoft Office (Open Office) OpenOffice.org and LibreOffice (Open Document), and Apple's iWork. XML has also been employed as the base language for communication protocols, such as XMPP.

Chapter 2

SYSTEM ANALYSIS

2.1 INTRODUCTION TO MINI PROJECT

PRODUCT DESCRIPTION: Library Management System is a computerized system which helps user (librarian) to manage the library daily activity in electronic format. It reduces the risk of paper work such as file lost, file damaged and time consuming. It can help user to manage the transaction or record more effectively and timesaving.

PROBLEM STATEMENT: The problem occurred before having computerized system includes:

- **File lost** when computerized system is not implemented file is always lost because of human environment. Sometimes due to some human error there may be a loss of records.
- **File damaged** when a computerized system is not there file is always lost due to some accident like spilling of water by some member on file accidentally. Besides some natural disaster like floods or fires may also damage the files.
- **Difficult to search record** when there is no computerized system there is always a difficulty in searching of records if the records are large in number.
- **Space consuming** After the number of records become large the space for physical storage of file and records also increases if no computerized system is implemented.
- **Cost consuming** As there is no computerized system there to add each record paper will be needed which will increase the cost for the management of library.

2.2 SYSTEM OBJECTIVES

- **Improvement in control and performance** the system is developed to cope up with the current issues and problems of library. The system can add user, validate user and is also bug free.
- **Save cost** After computerized system is implemented less human force will be required to maintain the library thus reducing the overall cost.
- **Save time** Librarian is able to search record by using few clicks of mouse and few search keywords thus saving his valuable time.

- **Option of online Notice board** Librarian will be able to provide a detailed description of workshops going in the college as well as in nearby colleges
- **Lecture Notes** Teacher have a facility to upload lectures notes in a pdf file having size not more than 10mb

2.3 SOFTWARE AND HARDWARE REQUIREMENTS

This section describes the software and hardware requirements of the system

SOFTWARE REQUIREMENTS

- **Operating system**- Windows 7 is used as the operating system as it is stable and supports more features and is more user friendly
- **MYSQL Database**-MYSQL is used as database as it easy to maintain and retrieve records by simple queries which are in English language which are easy to understand and easy to write.
- **Development tools and Programming language**- HTML is used to write the whole code and develop WebPages with CSS, java script for styling work and PHP for sever side scripting.

HARDWARE REQUIREMENTS

- Intel core i5 2nd generation is used as a processor because it is fast than other processors and provide reliable and stable and we can run our pc for longtime. By using this processor we can keep on developing our project without any worries.
- Ram 1 GB is used as it will provide fast reading and writing capabilities and will in turn support in processing

Chapter 3

SYSTEM DESIGN

3.1 TABLE DESIGN

VARIOUS TABELS TO MAINTAIN INFORMATION

Field	Datatype	Default	Key	Extra
Code	INT(11)	Not Null	Primary	Auto increment
Bookname	VARCHAR(255)	Null		
Author	VARCHAR(255)	Null		
Publication	VARCHAR(255)	Null		
Subject	VARCHAR(255)	Null		
No of copies	INT(10)	Null		

Table 3.1 BOOK TABLE FOR KEEPING TRACK OF BOOKS

Field	Data type	Default	Key	Extra
libid	INT(11)	NOT NULL	Primary key	Autoincrement
regno	INT(10)	NULL		
branch	VARCHAR(255)	NULL		
section	VARCHAR(255)	NULL		
semester	VARCHAR(255)	NULL		
section	VARCHAR(2)	NULL		
yearofadm	INT(5)	NULL		

Table 3.2 STUDENT TABLE FOR STUDENT INFORMATION

Field	Data Type	Default	Key	Extra
bookid	INT(11)	NOT NULL	Foreign key	References book
stuid	INT(11)	NOT NULL	Foreignkey	References Student
issuedate	DATE	NULL		
returndate	DATE	NULL		

Table 3.3 ISSUE TABLE TO KEEP TRACK OF BOOKS ISSUED

Field	Data type	Default	Key	Extra
logid	INT(11)	NOT NULL	Foreignkey	References Student
Username	VARCHAR(255)	NULL		
Password	VARCHAR(255)	NULL		
numbooks	INT(1)	NULL		

Table 3.4 STUDENT LOGIN TABLE

Field	Data type	Default	Key	Extra
Name	Varchar(255)	NULL		
Date	Date(yyyy/mm/dd)	NULL		
Time	VARCHAR(255)	NULL		
Mname	VARCHAR(255)	NULL		
Contactno.	Int(30)	NULL		
Email	VARCHAR(255)	NULL		
Venue	varchar(255)	NULL		

Table 3.5 EVENT TABLE FOR EVENT INFORMATION

- TEACHER LOGIN TABLE

Field	DataType	Default	Key	Extra
Loginid	INT(11)	NOT NULL	Foreign key	References teacher
Username	VARCHAR(255)	NULL		
Password	VARCHAR(255)	NULL		

3.2 DATA FLOW DIAGRAMS

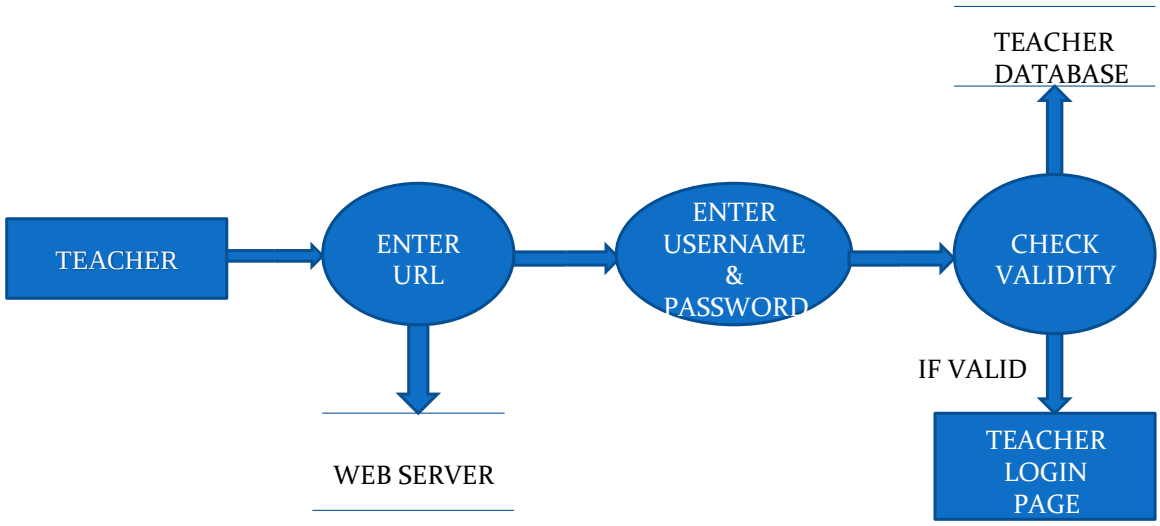


FIG 3.1 Data Flow Diagram For Teacher Login

After entering to the home page of the website, teacher can choose the TEACHER LOGIN option where they are asked to enter username & password, and if he/she is a valid user then a teacher login page will be displayed.

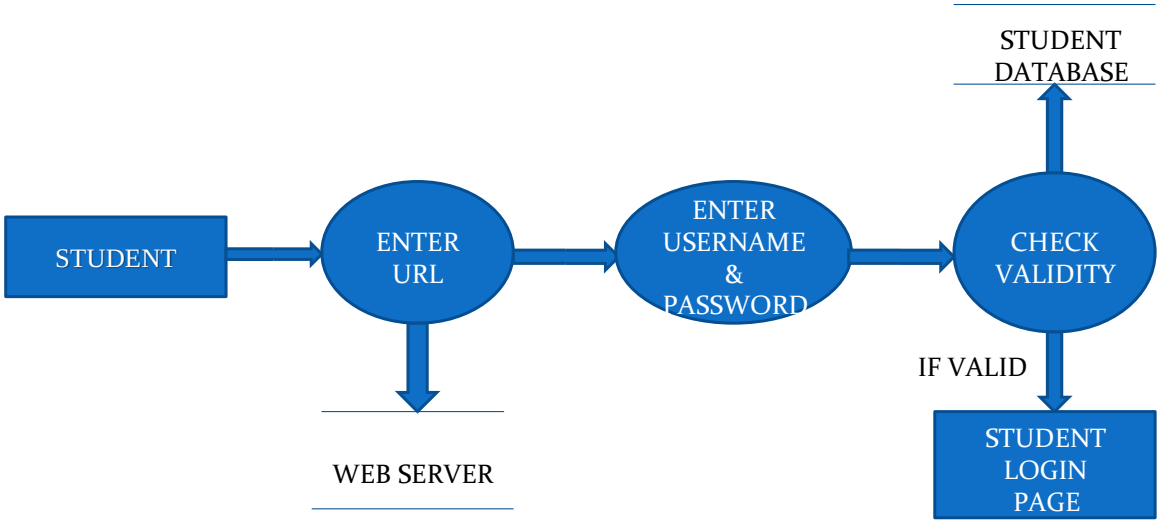


FIG 3.2 DATA FLOW DIAGRAM FOR STUDENT LOGIN

After entering to the home page of the website, student can choose the STUDENT LOGIN option where they are asked to enter username & password, and if he/she is a valid user then a student login page will be displayed.

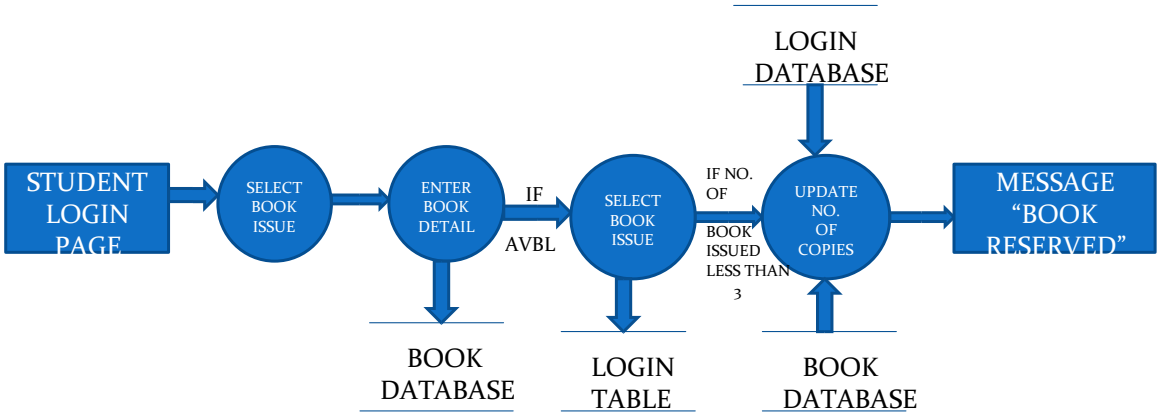
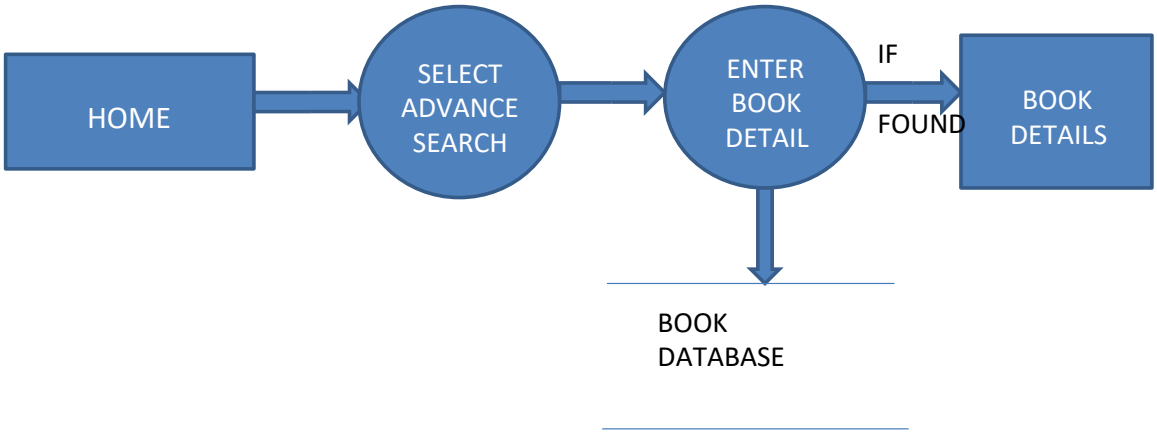


FIG 3.3 Data Flow Diagram For Book Issue

It is a 2nd level Data Flow Diagram where after entering STUDENT LOGIN page he/she can select a book issue option where after entering the book detail, he/she can select the book issue option and if the maximum no of books issued limit is not crossed then a request will be sent to the librarian who will approve the book issue.



3.4 Data Flow Diagram For Book Search

After the home page login there will be an option of the book search where after entering book detail like author name, publication, book name etc book details will be displayed.

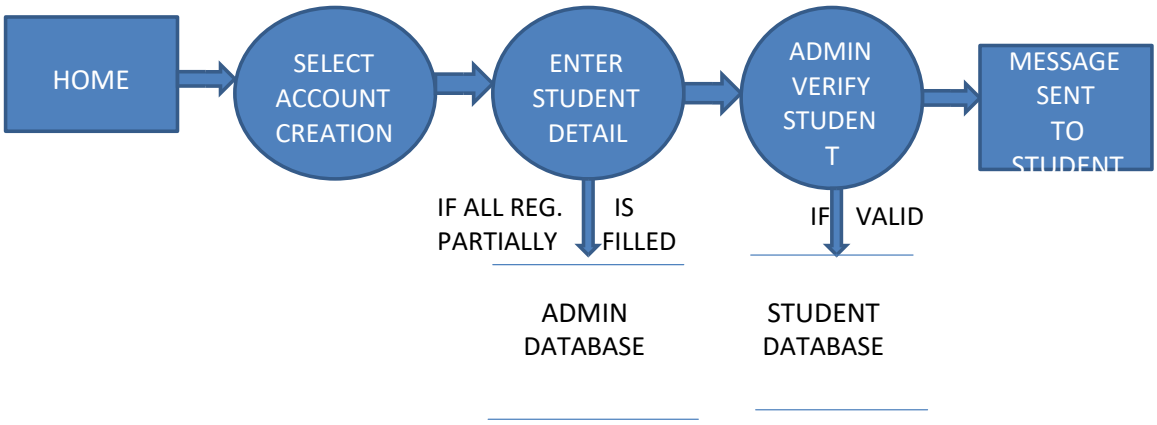


FIG 3.5 Data Flow Diagram For Account Creation

After the home page login there will be an option of CREATE AN ACCOUNT where after entering student detail, if all the fields are filled then a request will be sent to the librarian who will approve him as a registered member of the library.

Chapter 4

SYSTEM IMPLEMENTATION

4.1 SAMPLE CODE

Dashboard:

```
<?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;" /> <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>
```

Admin:

```
<?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;" /> <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>
```

4.2 Coding Description

We have used various web technologies for the front-end and back-end development, the following are the main technologies :

Front End: HTML, CSS, JavaScript, jquery, bootstrap

Back End: MySQL Database, PHP

HTML, HyperText Markup Language, gives content structure and meaning by defining that content as, for example, headings, paragraphs, or images. CSS, or Cascading Style Sheets, is a presentation language created to style the appearance of content—using, for example, fonts or colors. Node.js is an open-source, cross-platform, JavaScript runtime environment that executes JavaScript code outside of a browser.

Express.js is a Node js web application server framework, which is specifically designed for building single-page, multi-page, and hybrid web applications. It has become the standard server framework for node.js.

JavaScript, often abbreviated as JS, is a high-level, interpreted scripting language that conforms to the ECMAScript specification. JavaScript has curly-bracket syntax, dynamic typing, prototype-based object-orientation, and first-class functions.

MySQL is an open-source relational database management system.

Bootstrap is the most popular HTML, CSS, and JavaScript framework for developing responsive, mobile-first websites.

4.3 Software Testing

The aim of the system testing process was to determine all defects in our project.The rogram was subjected to a set of test inputs and various observations were made and based on these observations it will be decided whether the program behaves as expected or not.

UNIT TESTING

Unit testing is undertaken when a module has been created and successfully reviewed.In order to test a single module we need to provide a complete environment i.e. besides the module we would require

- The procedures belonging to other modules that the module under test calls
- Non local data structures that module accesses
- A procedure to call the functions of the module under test with appropriate parameters

Unit testing was done on each and every module that is described under module description of chapter 4

1. Test For the admin module

- Testing admin login form-This form is used for log in of administrator of the system.In this we enter the username and password if both are correct administration page will open other wise if any of data is wrong it will get redirected back to the login page and again ask for username and password
- Student account addition- In this section the admin can verify student details from student academinc info and then only add student details to main library database it contains add and delete buttons if user click add button data will be added to student database and if he clicks delete button the student data will be deleted

2. Book Addition- Admin can enter details of book and can add the details to the main book table also he can view the books requests. Test for Student login module

- Test for Student login Form-This form is used for log in of Student. In this we enter the libraryid, username and password if all these are correct student login page will open otherwise if any of data is wrong it will get redirected back to the login page and again ask for libraryid, username and password.

- Test for account creation- This form is used for new account creation when student does not fill the form completely it asks again to fill the whole form when he fill the form fully it gets redirected to page which show waiting for conformation message as his data will be only added by administrator after verification.

3. Test for teacher login module-

- Test for teacher login form- This form is used for login of teacher. In this we enter the username and password if all these are correct teacher login page will open otherwise if any of data is wrong it will get redirected back to the login page and again ask for username and password.

INTEGRATION TESTING

In this type of testing we test various integration of the project module by providing the input. The primary objective is to test the module interfaces in order to ensure that no errors are occurring when one module invokes the other module.

Chapter 5

OUTCOME OF THE PROJECT

5.1 Screenshots

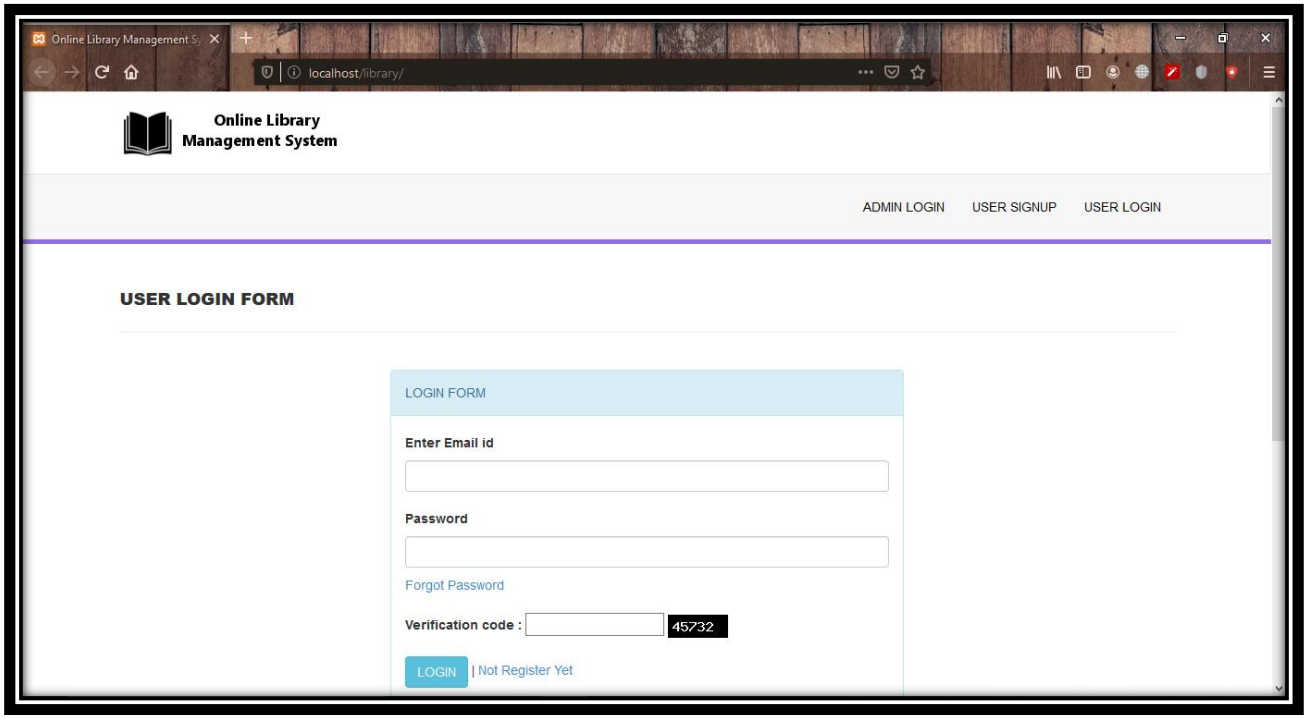


FIG 5.1 User login form

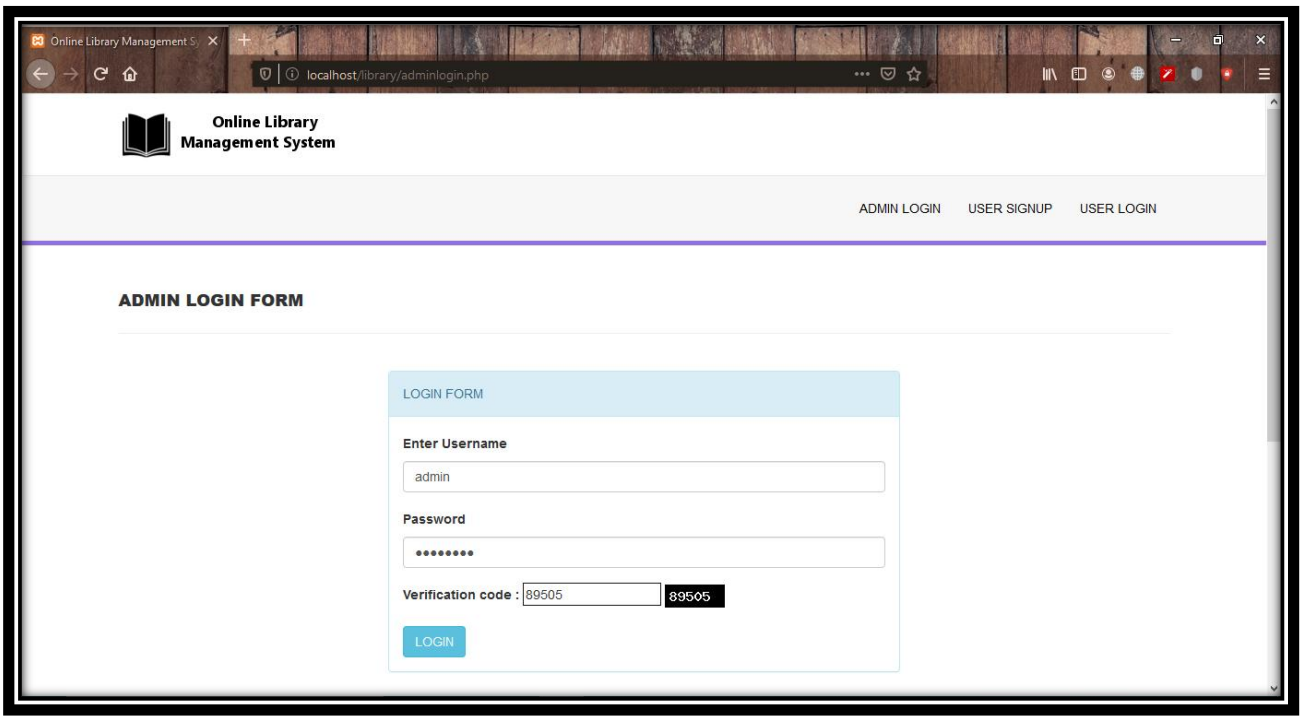


FIG 5.2 Admin Login Form

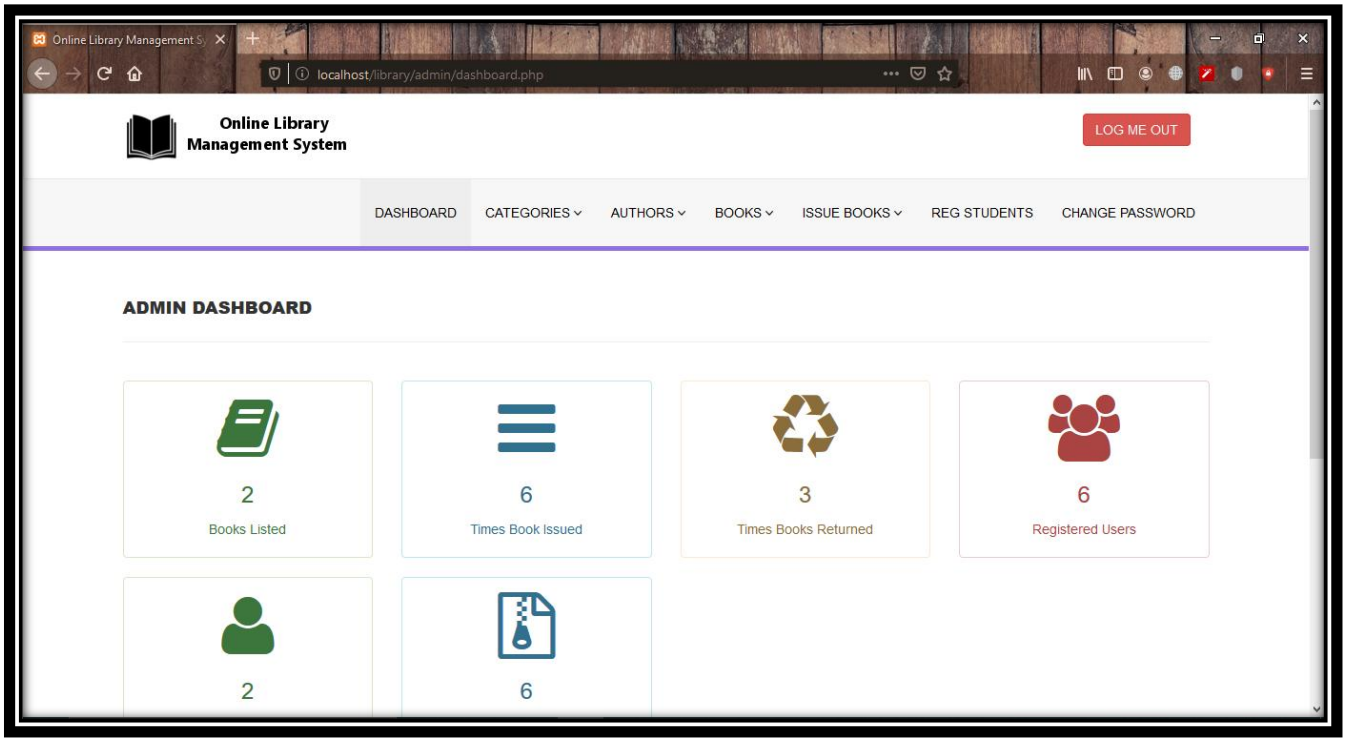


FIG 5.3 Admin dashboards

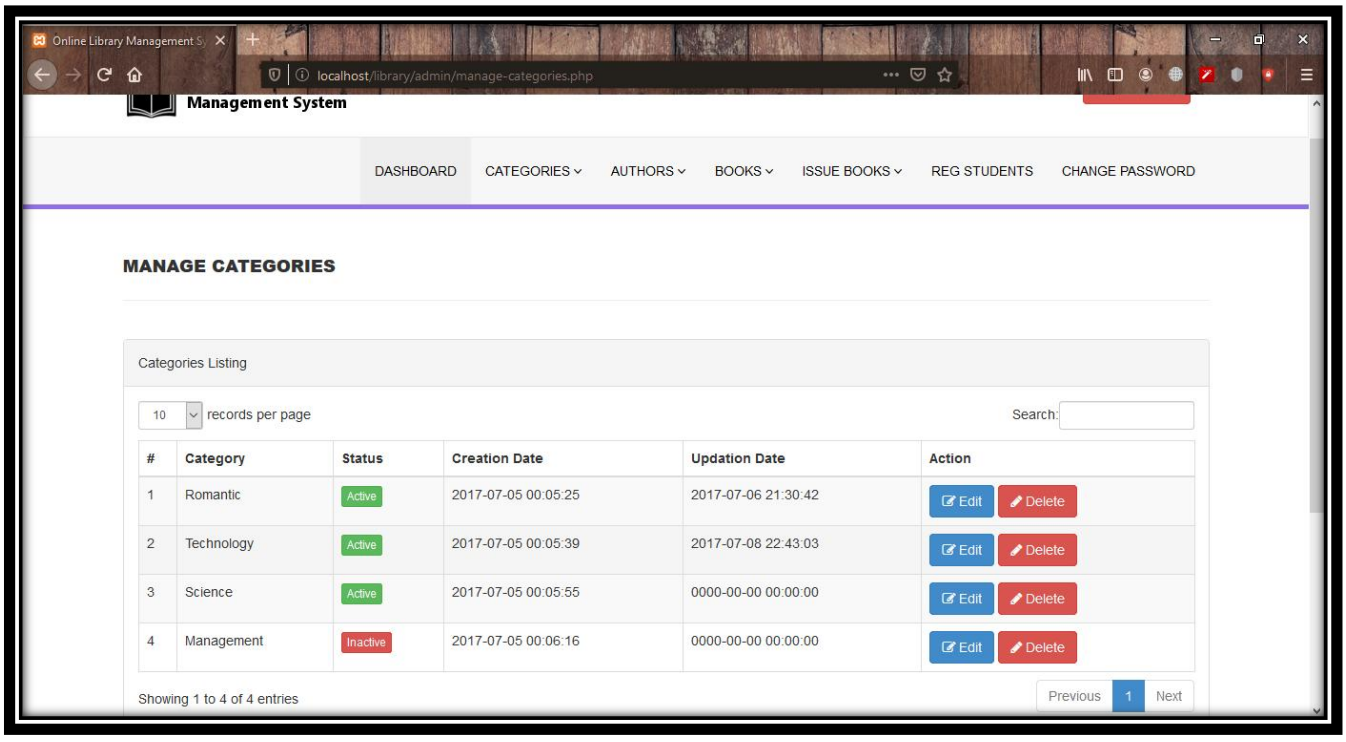


FIG 5.4 Manage Categories

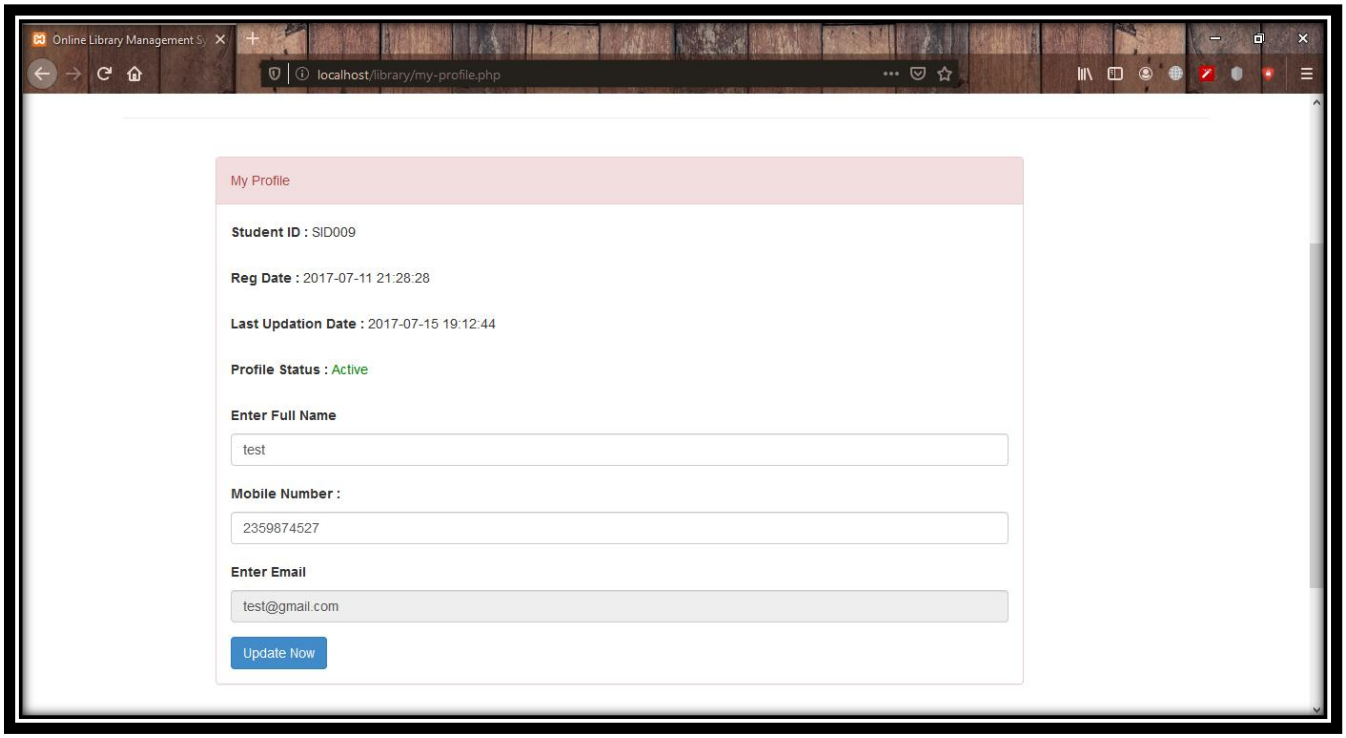


FIG 5.5 Student Profile

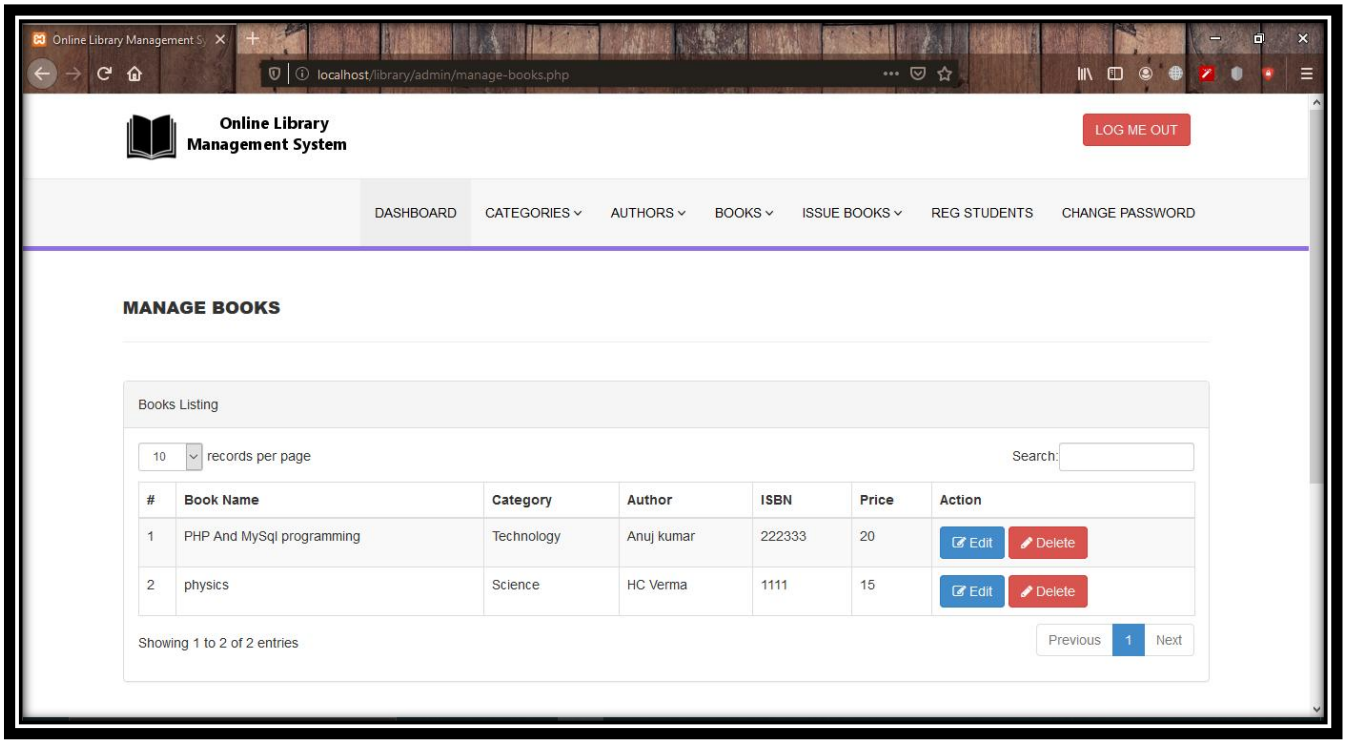


FIG 5.6 Manage Books

Chapter 6

CONCLUSION AND FUTURE ENHANCEMENTS

This website provides a computerized version of library management system, which will benefit the students, as well as the staff of the library.

It makes entire process online where student can search books; staff can generate reports and do book transactions. It also has a facility for student login where student can login and can see status of books issued as well request for book or give some suggestions. It has a facility of teacher's login where teachers can add lectures notes, also give necessary suggestion to library, and add info about workshops or events happening in our college or nearby college in the online notice board.

There is a future scope of this facility that many more features such as online lectures video tutorials can be added by teachers as well as online assignments submission facility, a feature Of group chat where students can discuss various issues of engineering can be added to this project thus making it more interactive more user friendly and project which fulfills each users need in the best way possible

REFERENCES

- http://www.w3schools.com/html/html_intro.asp
- http://www.w3schools.com/css/css_background.asp
- http://www.w3schools.com/js/js_datatypes.asp
- http://www.w3schools.com/sql/sql_insert.asp
- http://www.w3schools.com/sql/sql_update.asp
- http://www.w3schools.com/php/php_forms.asp
- Fundamentals of software engineering by Rajib mall, PHIlearning
- Web development and application development by Ivan Byross BPB publications