

LIBRARY MANAGEMENT SYSTEM

A MINI PROJECT REPORT

Submitted by

KONDAPALLI ABHIRAM
[RA2011003011133]
V ALLEN
JEROME[RA2011003011167]
JEYANTH
PRAKASH[RA2011003011166]

Under the guidance of

M. Karthikeyan
Assistant professor
Department of Data Science and Business Systems

In partial satisfaction of the requirements for the degree of

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE & ENGINEERING



SCHOOL OF COMPUTING
COLLEGE OF ENGINEERING AND TECHNOLOGY
SRM INSTITUTE OF SCIENCE AND TECHNOLOGY
KATTANKULATHUR – 603203

APRIL 2023



SRM INSTITUTE OF SCIENCE & TECHNOLOGY
COLLEGE OF ENGINEERING & TECHNOLOGY
S.R.M. NAGAR, KATTANKULATHUR – 603 203

BONAFIDE CERTIFICATE

Certified that this project report “**LIBRARY MANAGEMENT SYSTEM**” is the bonafide work of “**KONDAPALLI ABHIRAM[RA2011003011133], V ALLEN JEROME[RA2011003011167], JEYANTH PRAKASH[RA2011003011166]**” of III Year/VI Sem B.tech(CSE) who carried out the mini project work under my supervision for the course 18CSC303J- Database Management systems in SRM Institute of Science and Technology during the academic year 2022-2023(Even sem).

SIGNATURE

Mr. M. Karthikeyan
Assistant Professor
Department of Data Science and Science and
Buisness Systems
SRM Institute of Science and Technology

HOD Designation

Dr. M. Pushpalatha
Professor and HOD
Department of Computing Technologies
SRM Institute of Science and Technology

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 teachers 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 ie admin account can generate various reports such as student report , issue report, teacher report and book report.

Overall this 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 also reduce the human efforts.

TABLE OF CONTENTS

ABSTRACT	3
1. INTRODUCTION	5
1.1 PROJECT AIMS AND OBJECTIVES	5
1.2 BACKGROUND OF PROJECT	6
1.3 OPERATION ENVIRONMENT	7
2. SYSTEM ANALYSIS	8
2.1 SOFTWARE REQUIREMENT SPECIFICATION	9
2.2 EXISTING VS PROPOSED	14
2.3 SOFTWARE TOOL USED	15
3. SYSTEM DESIGN	19
3.1 TABLE DESIGN	20
3.2 DATA FLOW DIAGRAM'S	23
4. SYSTEM IMPLEMENTATION	28
4.1 MODULE DESCRIPTION	29
4.2 SCREEN SHOTS	36
5. SYSTEM TESTING	59
5.1 UNIT TESTING	59
5.2 INTEGRATION TESTING	61
6. CONCLUSION & FUTURE SCOPE	62
7. REFERENCES	63

CHAPTER 1

INTRODUCTION

This chapter gives an overview about the aim , objectives ,background and operation environment of the system.

1.1 PROJECT AIMS AND OBJECTIVES

The project aims and objectives that will be achieved after completion of this project are discussed in this subchapter. The aims and objectives are as follows:

- Online book issue
- Request column for librarian for providing new books
- A separate column for digital library
- Student login page where student can find books issued by him/her and date of return.
- A search column to search availability of books
- A teacher login page where teacher can add any events being organized in the college and important suggestions regarding books.
- Online notice board about the workshop.

1.2 BACKGROUND OF PROJECT

Library Management System is an application which refers to library systems which are generally small or medium in size. It is used by librarian to manage the library using a computerized system where he/she can record various transactions like issue of books, return of books, addition of new books, addition of new students etc.

Books and student maintenance modules are also included in this system which would keep track of the students using the library and also a detailed description about the books a library contains. With this computerized system there will be no loss of book record or member record which generally happens when a non computerized system is used.

In addition, report module is also included in Library Management System. If user's position is admin, the user is able to generate different kinds of reports like lists of students registered, list of books, issue and return reports.

All these modules are able to help librarian to manage the library with more convenience and in a more efficient way as compared to library systems which are not computerized.

1.5 OPERATION ENVIRONMENT

PROCESSOR	INTEL CORE PROCESSOR OR BETTER PERFORMANCE
OPERATING SYSTEM	WINDOWS VISTA ,WINDOWS7, UBUNTU
MEMORY	1GB RAM OR MORE
HARD DISK SPACE	MINIMUM 3 GB FOR DATABASE USAGE FOR FUTURE
DATABASE	MY SQL

CHAPTER 2

SYSTEM ANALYSIS

In this chapter, we will discuss and analyze about the developing process of Library Management System including software requirement specification (SRS) and comparison between existing and proposed system . The functional and non functional requirements are included in SRS part to provide complete description and overview of system requirement before the developing process is carried out. Besides that, existing vs proposed provides a view of how the proposed system will be more efficient than the existing one.

2.1 SOFTWARE REQUIREMENT SPECIFICATION

2.1.1 GENERAL DESCRIPTION

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 time-saving.

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. Some times 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 the to add each record paper will be needed which will increase the cost for the management of library.

2.1.2 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

2.1.3 SYSTEM REQUIREMENTS

2.1.3.1 NON FUNCTIONAL REQUIREMENTS

- Product Requirements

EFFICIENCY REQUIREMENT

When a library management system will be implemented librarian and user will easily access library as searching and book transaction will be very faster .

RELIABILITY REQUIREMENT

The system should accurately performs member registration ,member validation , report generation, book transaction and search

USABILITY REQUIREMENT

The system is designed for a user friendly environment so that student and staff of library can perform the various tasks easily and in an effective way.

ORGANIZATIONAL REQUIREMENT

IMPLEMENTATION REQUIREMENTS

In implementing whole system it uses html in front end with php as server side scripting language which will be used for database connectivity and the backend ie the database part is developed using mysql.

DELIVERY REQUIREMENTS

The whole system is expected to be delivered in six months of time with a weekly evaluation by the project guide.

2.1.3.2 FUNCTIONAL REQUIREMENTS

1. NORMAL USER

1.1 USER LOGIN

Description of feature

This feature used by the user to login into system. They are required to enter user id and password before they are allowed to enter the system. The user id and password will be verified and if invalid id is there user is allowed to not enter the system.

Functional requirements

- user id is provided when they register
- The system must only allow user with valid id and password to enter the system
- The system performs authorization process which decides what user level can access to.
- The user must be able to logout after they finished using system.

1.2 REGISTER NEW USER

Description of feature

This feature can be performed by all users to register new user to create account.

Functional requirements

- System must be able to verify information
- System must be able to delete information if information is wrong

1.3 REGISTER NEW BOOK

Description of feature

This feature allows to add new books to the library

Functional requirements

- System must be able to verify information
- System must be able to enter number of copies into table.
- System must be able to not allow two books having same book id.

1.5 SEARCH BOOK

DESCRIPTION OF FEATURE

This feature is found in book maintenance part . we can search book based on book id , book name , publication or by author name.

Functional requirements

- System must be able to search the database based on select search type
- System must be able to filter book based on keyword entered
- System must be able to show the filtered book in table view

1.5 ISSUE BOOKS AND RETURN BOOKS

DESCRIPTION OF FEATURE

This feature allows to issue and return books and also view reports of book issued.

Functional requirements

- System must be able to enter issue information in database.
- System must be able to update number of books.
- System must be able to search if book is available or not before issuing books
- System should be able to enter issue and return date information

1.6 EVENT ADDITION

DESCRIPTION OF FEATURE

This feature allows teacher and student to add information about various workshops being conducted in college and colleges nearby.

Functional requirements

- System should be able to add detailed information about events .
- System should be able to display information on notice board available in the homepage of site

2.1.4 SOFTWARE AND HARDWARE REQUIREMENTS

This section describes the software and hardware requirements of the system

2.1.4.1 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
- Database MYSQL-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.

2.1.4.2 HARDWARE REQUIREMENTS

- Intel core i5 2nd generation is used as a processor because it is fast than other processors an 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

2.2 EXISTING VS PROPOSED SYSTEM

- i. Existing system does not have any facility of teachers login or student login where as proposed system will have a facility of student login as well as teacher's login
- ii. Existing system does not have a facility of online reservation of books whereas proposed system has a facility of online reservation of books
- iii. Existing system does not have any facility of online notice board where description of workshops happening in our college as well as nearby colleges is being provided.
- iv. Existing system does not has any option of lectures notes uploaded by teachers whereas proposed system will have this facility
- v. Existing system does not have any facility to generate student reports as well book issue reports whereas proposed system provides librarian with a tool to generate reports
- vi. Existing system does not has any facility for book request and sugeestions where as in proposed system after logging in to their accounts student can request books as well as provide suggestions to improve library

2.3 SOFTWARE TOOLS USED

The whole Project is divided in two parts the front end and the back end.

2.3.1 Front end

The front end is designed using of html , Php ,css, Java script

- **HTML- HTML or Hyper Text Markup Language** is the main markup language for creating web pages and other information that can be displayed in a web browser. HTML is written in the form of HTML elements consisting of *tags* enclosed in angle brackets (like `<html>`), within the web page content. HTML tags most commonly come in pairs like `<h1>` and `</h1>`, although some tags represent *empty elements* and so are unpaired, for example ``. The first tag in a pair is the *start tag*, and the second tag is the *end tag* (they are also called *opening tags* and *closing tags*). In between these tags web designers can add text, further tags, comments and other types of text-based content. The purpose of a web browser is to read HTML documents and compose them into visible or audible web pages. The browser does not display the HTML tags, but uses the tags to interpret the content of the page. HTML elements form the building blocks of all websites. HTML allows images and objects to be embedded and can be used to create interactive forms. It provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items. It can embed scripts written in languages such as JavaScript which affect the behavior of HTML web pages.
- **CSS- Cascading Style Sheets (CSS)** is a style sheet language used for describing the look and formatting of a document written in a markup language. While most often used to style web pages and interfaces written in HTML and XHTML, the language can be applied to any kind of XML document, including plain XML, SVG and XUL. CSS is a cornerstone specification of the web and almost all web pages use CSS style sheets to describe their presentation. CSS is designed primarily to enable the separation of document content from document presentation, including elements such as the layout, colors, and fonts. This separation can improve content accessibility, provide more flexibility and control in the specification

of presentation characteristics, enable multiple pages to share formatting, and reduce complexity and repetition in the structural content (such as by allowing for table less web design).CSS can also allow the same markup page to be presented in different styles for different rendering methods, such as on-screen, in print, by voice (when read out by a speech-based browser or screen reader) and on Braille-based, tactile devices. It can also be used to allow the web page to display differently depending on the screen size or device on which it is being viewed. While the author of a document typically links that document to a CSS file, readers can use a different style sheet, perhaps one on their own computer, to override the one the author has specified. However if the author or the reader did not link the document to a specific style sheet the default style of the browser will be applied.CSS specifies a priority scheme to determine which style rules apply if more than one rule matches against a particular element. In this so-called *cascade*, priorities or *weights* are calculated and assigned to rules, so that the results are predictable.

- **JAVA SCRIPT- JavaScript (JS)** is a dynamic computer programming language. It is most commonly used as part of web browsers, whose implementations allow client-side scripts to interact with the user, control the browser, communicate asynchronously, and alter the document content that is displayed. It is also being used in server-side programming, game development and the creation of desktop and mobile applications. JavaScript is a prototype-based scripting language with dynamic typing and has first-class functions. Its syntax was influenced by C. JavaScript copies many names and naming conventions from Java, but the two languages are otherwise unrelated and have very different semantics. The key design principles within JavaScript are taken from the Self and Scheme programming languages. It is a multi-paradigm language, supporting object-oriented, imperative, and functional programming styles. The application of JavaScript to use outside of web pages—for example, in PDF documents, site-specific browsers, and desktop widgets—is also significant. Newer and faster JavaScript VMs and platforms built upon them (notably Node.js) have also increased the popularity of JavaScript for server-side web applications. On the client side, JavaScript was traditionally implemented as

an interpreted language but just-in-time compilation is now performed by recent (post-2012) browsers.

- PHP- **PHP** is a server-side scripting language designed for web development but also used as a general-purpose programming language. PHP is now installed on more than 244 million websites and 2.1 million web servers. Originally created by Rasmus Lerdorf in 1995, the reference implementation of PHP is now produced by The PHP Group. While PHP originally stood for *Personal Home Page*, it now stands for *PHP: Hypertext Preprocessor*, a recursive backronym. PHP code is interpreted by a web server with a PHP processor module, which generates the resulting web page: PHP commands can be embedded directly into an HTML source document rather than calling an external file to process data. It has also evolved to include a command-line interface capability and can be used in standalone graphical applications. PHP is free software released under the PHP License. PHP can be deployed on most web servers and also as a standalone shell on almost every operating system and platform, free of charge.

2.3.2 BACK END- The back end is designed using mysql which is used to design the databases

- **MYSQL- MySQL** ("My S-Q-L", officially, but also called "My Sequel") is (as of July 2013) the world's second most widely used open-source relational database management system (RDBMS). It is named after co-founder Michael Widenius daughter, My. The SQL phrase stands 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 .MySQL is a popular choice of database for use in web applications, and is a central component of the widely used LAMP open source web application software stack (and other 'AMP' stacks). LAMP is an acronym for "Linux, Apache, MySQL, Perl/PHP/Python." Free-software-open source projects that require a full-featured database management system often use MySQL. For commercial use, several paid editions are available, and offer additional functionality. Applications which use MySQL databases

include: TYPO3, MODx, Joomla, WordPress, phpBB, MyBB, Drupal and other software. MySQL is also used in many high-profile, large-scale websites, including Wikipedia, Google (though not for searches), Facebook, Twitter, Flickr, and YouTube

CHAPTER 3

SYSTEM DESIGN

3.1 TABLE DESIGN

VARIOUS TABLES TO MAINTAIN INFORMATION

- BOOK TABLE FOR KEEPING TRACK OF BOOKS

Field	Data type	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		

● STUDENT TABLE FOR STUDENT INFORMATION

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

● TEACHER TABLE TO KEEP TEACHER INFORMATION

Field	Data Type	Default	Key	Extra
Tid	INT(11)	NOT NULL	Primary key	Auto increment
Name	VARCHAR(255)	NULL		
Designation	VARCHAR(255)	NULL		
Branch	VARCHAR(255)	NULL		
Contactno	INT(13)	NULL		
Lectures	LONG BLOB	NULL		

- Issue table to keep track of books issued

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

- STUDENT LOGIN TABLE

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

● EVENT TABLE FOR EVENT INFORMATION

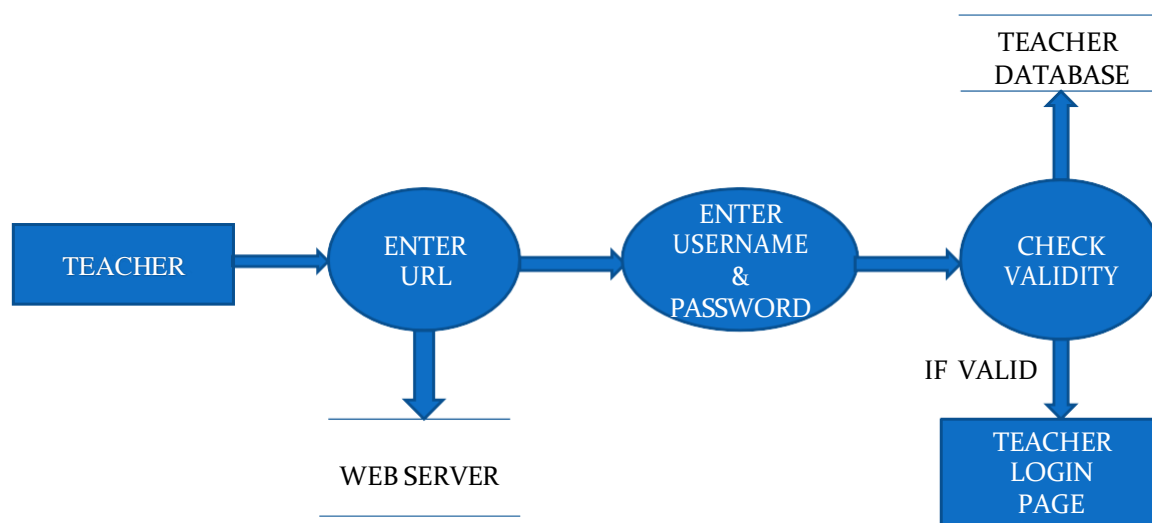
Field	Data type	Default	Key	Extra
Name	Varchar(255)	NULL		
Date	Date(yyyy/mm/dd)	NULL		
Time	VARCHAR(255)	NULL		
Mname	VARCHAR(255)	NULL		
Contactno.	Int(30)	NULL		
Email	VARCHAR(255)	NULL		
Venue	varchar(255)	NULL		

● TEACHER LOGIN TABLE

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

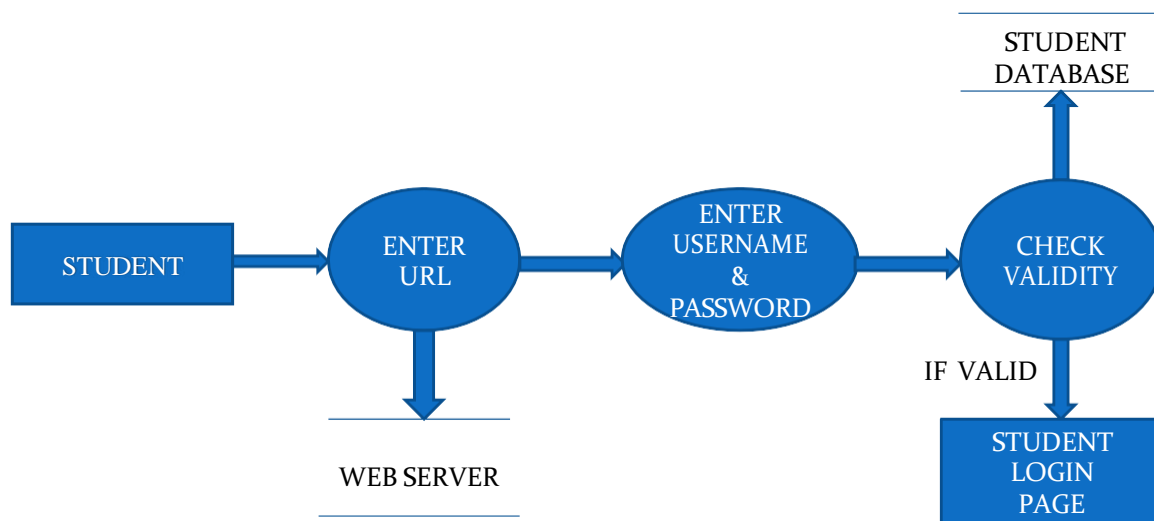
3.2 DATA FLOW DIAGRAMS

DATA FLOW DIAGRAM FOR TEACHER LOGIN



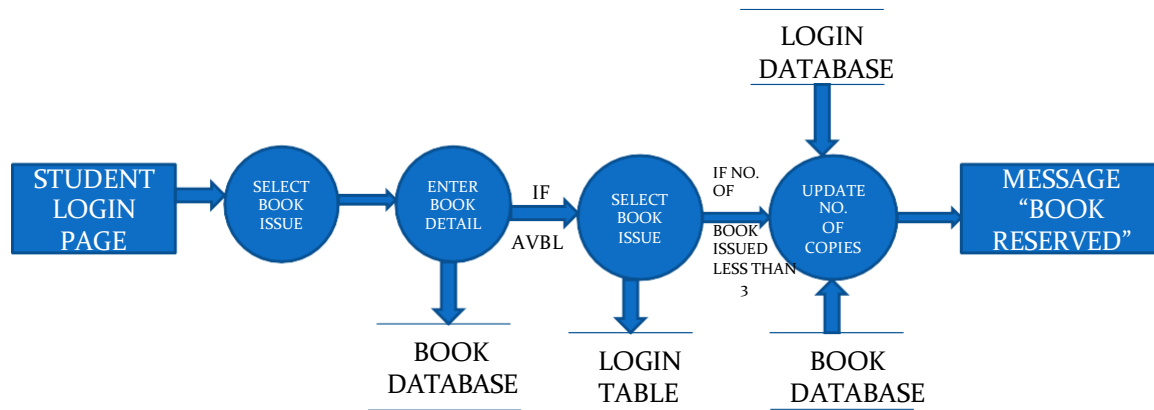
After entering to the home page of the website , teacher can choose the TEACHER LOGIN option where they are asked to enter username & password , and if he/she is a valid user then a teacher login page will be displayed.

DATA FLOW DIAGRAM FOR STUDENT LOGIN



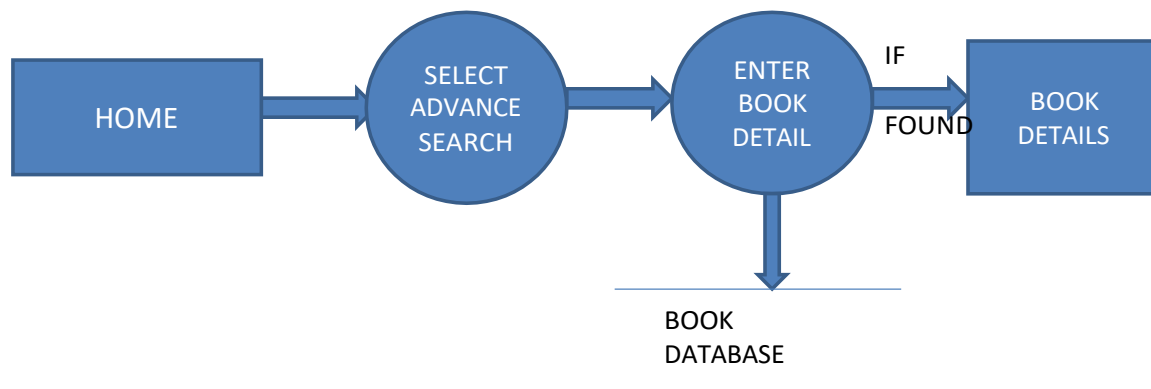
After entering to the home page of the website , student can choose the STUDENT LOGIN option where they are asked to enter username & password , and if he/she is a valid user then a student login page will be displayed.

DATA FLOW DIAGRAM FOR BOOK ISSUE



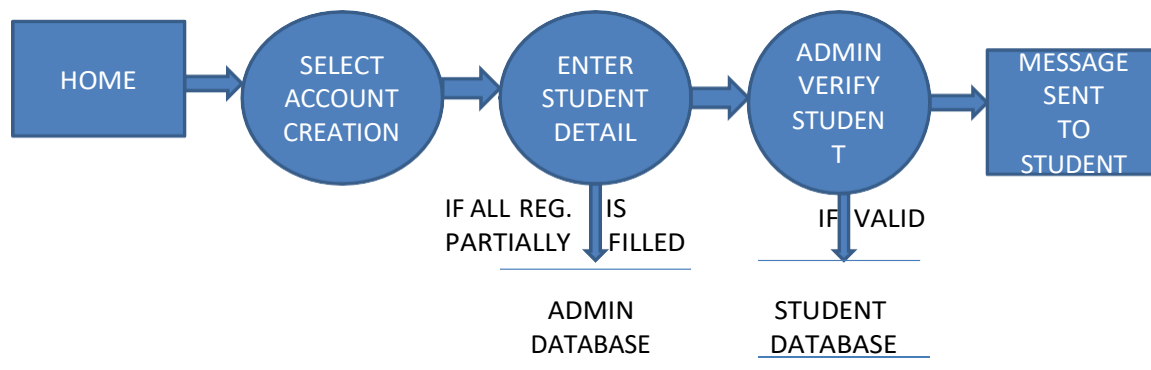
It is a 2nd level Data Flow Diagram where after entering STUDENT LOGIN page he/she can select a book issue option where after entering the book detail, he/she can select the book issue option and if the maximum no of books issued limit is not crossed then a request will be sent to the librarian who will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

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.

DATA FLOW DIAGRAM FOR ACCOUNT CREATION

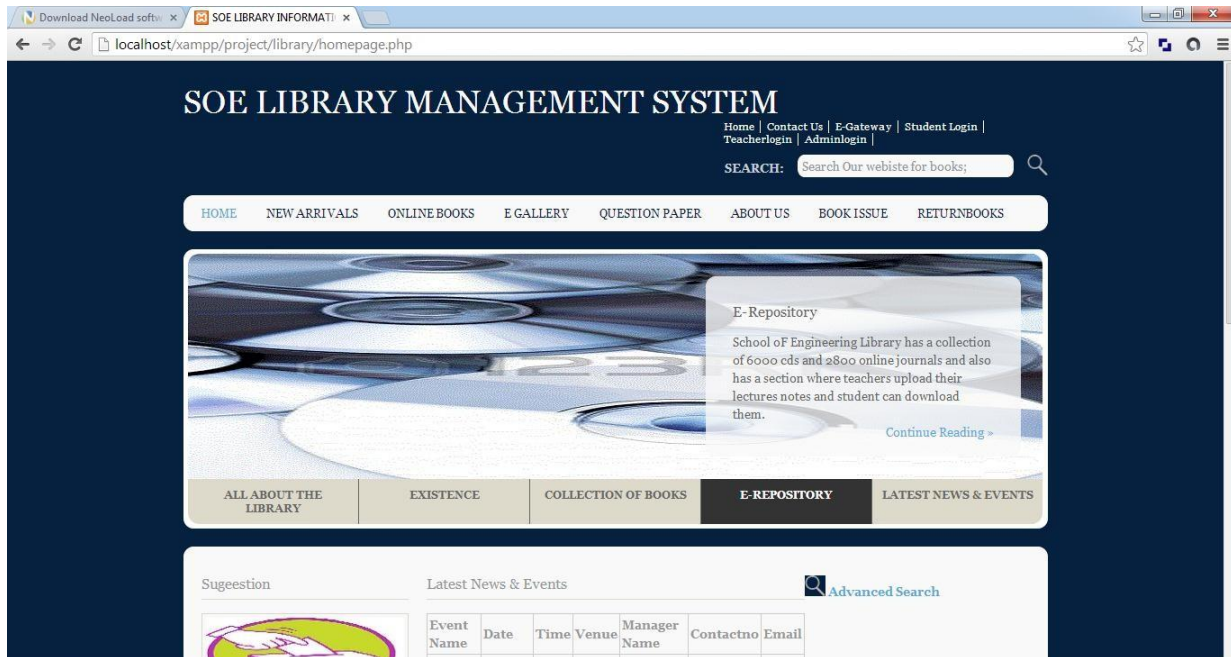


After the home page login there will be an option of CREATE AN ACCOUNT where after entering student detail ,if all the fields are filled then a request will be sent to the librarian who will approve him as a registered member of the library.

CHAPTER 4

SYSTEM IMPLEMENTATION

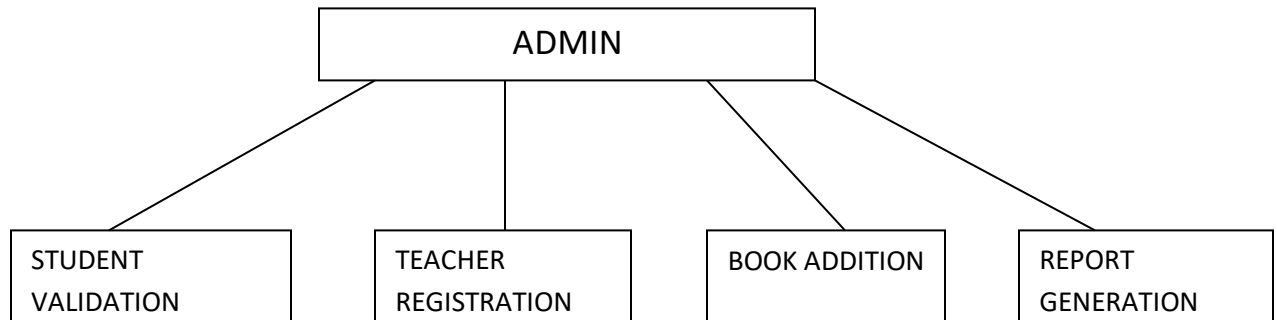
4.1.1 Screenshot for homepage



4.1 MODULE DESCRIPTION

For Library Management System it is divided into the following Modules:

4.1.1 Admin Module



The following module contains various facilities like student validation, teacher registration, book addition, and report generation.

4.1.1.1 Code for Admin module

4.1.1.1.1 Code for entering admin username and password

```
<?php
//Start session
session_start();
//Unset the variables stored in session
unset($_SESSION['SESS_FIRST_NAME']);
unset($_SESSION['SESS_LAST_NAME']);
?>
<!DOCTYPE html>
<html xml:lang="EN" lang="EN" dir="ltr">
<head>
<title>SOE LIBRARY INFORMATION SYSTEM</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
<meta http-equiv="imagetoolbar" content="no" />
<link rel="stylesheet" href="../styles/layout.css" type="text/css" />
<!-- Homepage Specific Elements -->
<script type="text/javascript" src="../scripts/jquery-1.4.1.min.js"></script>
<script type="text/javascript" src="../scripts/jquery-ui-1.7.2.custom.min.js"></script>
<script type="text/javascript" src="../scripts/jquery.tabs.setup.js"></script>
<!-- End Homepage Specific Elements -->
</head>
<body id="top">
<div class="wrapper row1">
  <div id="header" class="clear">
    <div class="fl_left">
```

```

        <p class="readmore"><a href="#">Continue Reading &raquo;</a></p>
    </div>
</div>
<div class="featured_box" id="fc4">
    <div class="floater">
        <h2>E-Repository</h2>
        <p>School oF Engineering Library has a collection of 6000 cds and 2800 online
journals and also has a section where teachers upload their lectures notes and student
can download them.</p>
        <p class="readmore"><a href="#">Continue Reading &raquo;</a></p>
    </div>
</div>
<div class="featured_box" id="fc5">
    <div class="floater">
        <h2>Latest News and events</h2>
        <p></p>
        <p class="readmore"><a href="#">Click here to go to latest news page;</a></p>
    </div>
</div>
<div id="featured_tabs">
    <li><a href="#fc1">All About The Library</a></li>
    <li><a href="#fc2">Existence</a></li>
    <li><a href="#fc3">Collection of Books</a></li>
    <li><a href="#fc4">E-Repository</a></li>
    <li class="last"><a href="#fc5">Latest News & Events</a></li>
</ul>
<div class="overlay_right"></div>
<!-- ##### -->
</div>
</div>

<!--
#####
##### -->
<div class="wrapper row3">
    <div class="rnd">
        <div id="container" class="clear">
            <!--
#####
##### -->
            <div id="homepage" class="clear">
                <!-- ##### -->
                <div id="left_column">

```

```

        <li><a href="http://localhost/xampp/project/library/newarrival.php">&raquo;
New Arrivals</a></li>
        <li><a href="http://localhost/xampp/project/library/onlinebooks.php">&raquo;
Online Books</a></li>
        <li><a href="http://localhost/xampp/project/library/egallery.php">&raquo; E-
Gallery</a></li>
        <li><a href="http://localhost/xampp/project/library/aboutus.php">&raquo;
About Us</a></li>
        <li><a href="http://localhost/xampp/project/library/contactus.php">&raquo;
Contact Us</a></li>
        <li><a href="http://localhost/xampp/project/library/questionpaper.php">&raquo;
Question Papers</a></li>
        <li><a href="http://localhost/xampp/project/library/latestnews.php">&raquo;
Latest Events & News</a></li>
    </ul>
</div>

<!--
#####
##### -->
</div>
</div>
</div>
<!--
#####
##### -->

<!--
#####
##### -->
<div class="wrapper">
    <div id="copyright" class="clear">
        <p class="fl_left">Copyright &copy; 2014 - All Rights Reserved for SOE
LIBRARY-</p>
    </div>
</div>
</body>
</html>

```

4.1.1.2 Code for checking of admin username and password information

```

<?php
    session_start();
    $host="localhost"; // Host name
    $username="root"; // Mysql username
    $password=""; // Mysql password

```

```

$db_name="admin"; // Database name
$tbl_name="adminlogin"; // Table name

// Connect to server and select database.
mysql_connect("$host", "$username", "$password")or die("cannot connect");
mysql_select_db("$db_name")or die("cannot select DB");

// username and password sent from form
$myusername=$_POST['username'];
$mypassword=$_POST['password'];
$sql="SELECT * FROM $tbl_name WHERE username='$myusername' and
password='$mypassword'";
$result=mysql_query($sql);

// Mysql_num_row is counting table row
$count=mysql_num_rows($result);

// If result matched $myusername and $mypassword, table row must be 1 row

if($count>0)
{
    session_regenerate_id();
    $member = mysql_fetch_assoc($result);
    $_SESSION['SESS_FIRST_NAME'] = $member['username'];
    $_SESSION['SESS_LAST_NAME'] = $member['password'];
    session_write_close();
    header("location: adminlogin1.php");
    exit();
}else {
//Login failed
$message_arr[] = 'user name and password not found';
$errorflag = true;
if($errorflag) {
    $_SESSION['ERRMSG_ARR'] = $message_arr;
    session_write_close();
    header("location: adminlogin.php");
    exit();
}
}
?>

```

4.1.1.3 Code for student validation

```

<?php
require '../include/connection.php';
    $result = mysqli_query($con,"SELECT * FROM student");
    while($row = mysqli_fetch_array($result))
    {
        $libid=$row['libid'] ;
    }
}

```



```

        $regno= $row['regno'] ;
        $name= $row['stuname'] ;
        $branch= $row['branch'] ;
        $semester= $row['semester'] ;
        $section= $row['section'] ;
        $yearofadm= $row['yearofadm'];
                $email=$row['email'];
                $username= $row['username'];
                $password= $row['password'];
    }
        mysqli_close($con);
?>
<?php
require '../include/connection1.php';
$admin="admin@soelibrary.com";
$sql="INSERT INTO student(Libid,regno,stuname,branch,semester,section,yearofadm)
VALUES
('$libid','$regno','$name','$branch','$semester','$section','$yearofadm')";
$result="INSERT INTO studentlogin(logid,username,password,numbooks)
VALUES
('$libid','$username','$password','')";
if (!mysqli_query($connection,$sql))
{
    die('Error1: ' . mysqli_error($connection));
}
else if (!mysqli_query($connection,$result))
{
    die('Error2: ' . mysqli_error($connection));
}
else
    mail($email,"registrationdetails","you are registered pls login with ur email id and
password","From: $admin\n");
echo"added 1 record";
mysqli_close($connection);
?>
<?php
require '../include/connection.php';
        $result = mysqli_query($con,"SELECT * FROM student");
        while($row = mysqli_fetch_array($result))
        {
            $libid=$row['libid'] ;
            $regno= $row['regno'] ;
            $name= $row['stuname'] ;
            $branch= $row['branch'] ;
            $semester= $row['semester'] ;
            $section= $row['section'] ;

```

```

        $yearofadm= $row['yearofadm'];
        $email=$row['email'];
        $username= $row['username'];
        $password= $row['password'];
    }
    $sql=mysqli_query($con,"DELETE FROM student WHERE libid='$libid'");
    mysqli_close($con);
?>

```

4.1.1.4 Code For teacher registration

```

<?php
$con=mysqli_connect("localhost","root","rahul#1991","library");
// Check connection
if (mysqli_connect_errno())
{
    echo "Failed to connect to MySQL: " . mysqli_connect_error();
}
$loginid=$_POST['teacherid'];
$username=$_POST['username'];
$password=$_POST['password'];
$teachername=$_POST['teachername'];
$designation=$_POST['designation'];
$branch=$_POST['branch'];
$contactno=$_POST['contact'];
$lecture="";
$sql="INSERT INTO teacher (Tid, name, designation, branch, contactno,lectures)
VALUES
( $loginid,$teachername,$designation,$branch,$contactno,$lecture)";
$sql="INSERT INTO teacherlogin ($loginid,$username,$password)
VALUES
( $loginid,$username, $password)";
if (!mysqli_query($con,$sql))
{
    echo "error";
}
if (!mysqli_query($con,$sql))
{
    echo "error1";
}
echo "successs";

mysqli_close($con);

```

?>

4.1.1.5 Code for book addition

```
<?php
$con=mysqli_connect("localhost","root","","library");
// Check connection
if (mysqli_connect_errno())
{
    echo "Failed to connect to MySQL: " . mysqli_connect_error();
}

$sql="INSERT INTO books (code, bookname, author, publication, subject,
numberofbooks)
VALUES
('$ _POST[bookid]','$ _POST[bookname]','$ _POST[author]','$ _POST[publication]','$ _P
OST[subject]','$ _POST[numberofbooks]')";

if (!mysqli_query($con,$sql))
{
    echo "error";
}
echo "successs";

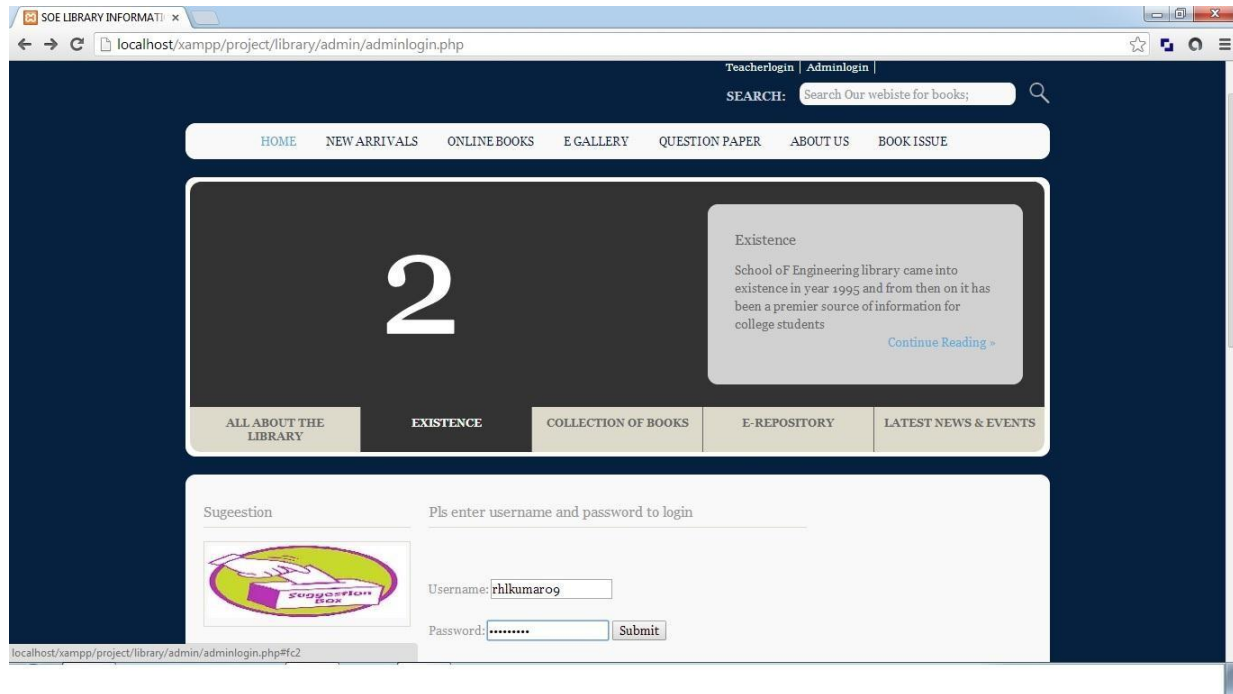
mysqli_close($con);
?>
<?php
$con=mysqli_connect("localhost","root","","admin");
// Check connection
if (mysqli_connect_errno())
{
    echo "Failed to connect to MySQL: " . mysqli_connect_error();
}

$sql="INSERT INTO newarrivals (code, bookname, author, publication, subject,
numberofbooks,arrivaldate)
VALUES
('$ _POST[bookid]','$ _POST[bookname]','$ _POST[author]','$ _POST[publication]','$ _P
OST[subject]','$ _POST[numberofbooks]','$ _POST[arrivaldate]')";

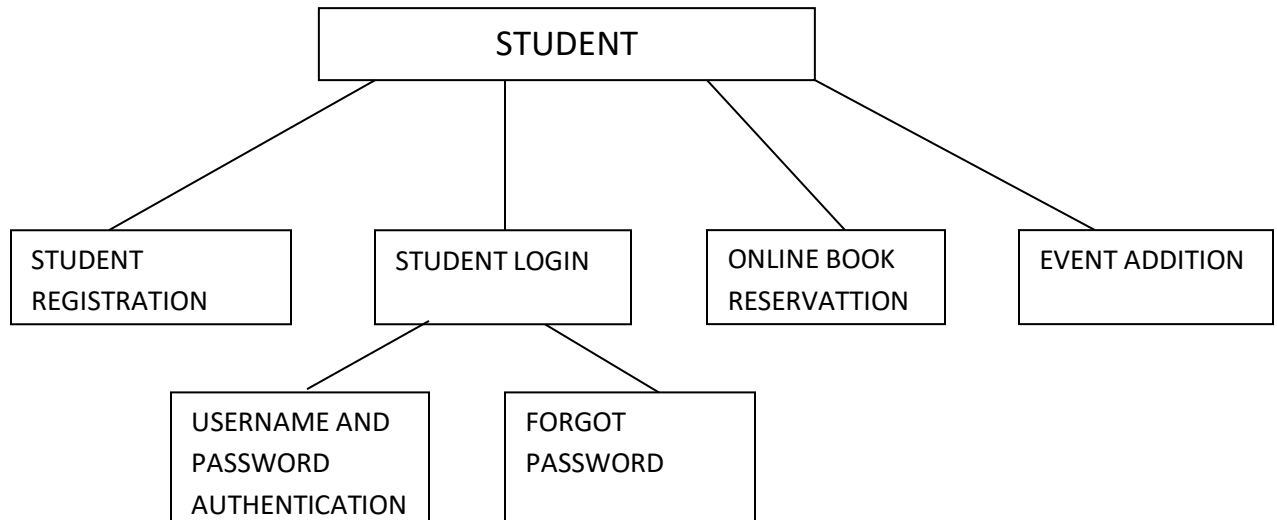
if (!mysqli_query($con,$sql))
{
    echo "error";
}
echo "successs";
mysqli_close($con);
```

?>

4.1.1 Screenshot for Admin login



4.1.2 Student Module



The following module contains various facilities like student registration, student login, online book reservation, and event addition. Any student if at any moment forgets his password he can retrieve it from forgot password option.

4.1.2.1 Code For Student account creation

```
<!DOCTYPE html>
<html xml:lang="EN" lang="EN" dir="ltr">
<head>
<title>SOE LIBRARY INFORMATION SYSTEM</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
<meta http-equiv="imagetoolbar" content="no" />
<link rel="stylesheet" href="../../styles/layout.css" type="text/css" />
<!-- Homepage Specific Elements -->
<script type="text/javascript" src="../../scripts/jquery-1.4.1.min.js"></script>
<script type="text/javascript" src="../../scripts/jquery-ui-1.7.2.custom.min.js"></script>
<script type="text/javascript" src="../../scripts/jquery.tabs.setup.js"></script>
<!-- End Homepage Specific Elements -->
</head>
<body id="top">
<div class="wrapper row1">
```

```

</tr>
<tr>
<td>Semester: </td> <td><select name="semester" size="1" tabindex="8">
<option selected="selected">1&2</option>
<option>3</option>
<option>4</option>
<option>5</option>
<option>6</option>
<option>7</option>
<option>8</option>
</select>
<span class="error">*</span><br><br></td>
</tr>
<tr>
<td>Section: </td> <td><select name="section" size="1" tabindex="2">
<option selected="selected">A</option>
<option>B</option>
</select>
<span class="error">*</span><br><br></td>
</tr>
<tr>
<td>Year of adm:</td> <td> <input type="text" name="yearofadm" size="30">
<span class="error">*</span><br><br></td>
</tr>
<tr>
<td>Email:</td> <td> <input type="email" name="email" size="30">
<span class="error">*</span><br><br></td>
</tr>

<tr>
<td>Username:</td> <td> <input type="text" name="username" size="30">
<span class="error">*</span><br><br></td>
</tr>
<tr>
<td>Password:</td> <td> <input type="password" name="password" size="30">
<span class="error">*</span><br><br></td>
</tr>
<tr>
<td><p>Click the submit</p></td>
</tr>
<tr>
<td height="40"><input type="submit" name="submit" value="submit" action=
"insertaccount.php"></td>
</tr>
</form>
</table>

```

```

if (empty($_POST["yearofadm"]))
    { $yearofadmErr = "year of adm. is required";
      echo $yearofadmErr;
      include("createaccount.php");
    }
else
    { $yearofadm = test_input($_POST["yearofadm"]);

      }
if (empty($_POST["username"]))
    { $usernameErr = "username is required";
      echo $usernameErr;
      include("createaccount.php");
    }
else
    { $username = test_input($_POST["username"]);

      }
if (empty($_POST["password"]))
    { $passwordErr = "password is required";
      echo $passwordErr;
      include("createaccount.php");
    }
else
    { $password = test_input($_POST["password"]); }

}

function test_input($data)
{
    $data = trim($data);
    $data = stripslashes($data);
    $data = htmlspecialchars($data);
    return $data;
}
?>
<?php
$con=mysqli_connect("localhost","root","","admin");
// Check connection
if (mysqli_connect_errno())
{
    echo "Failed to connect to MySQL: " . mysqli_connect_error();
}

```

```

$sql="INSERT INTO
student(libid,regno,stuname,branch,semester,section,yearofadm,email,username,password)
VALUES
('$_POST[libid]','$_POST[regno]','$_POST[stuname]','$_POST[branch]','$_POST[semester]','$_POST[section]','$_POST[yearofadm]','$_POST[email]','$_POST[username]','$_POST[password]');
if (!mysqli_query($con,$sql))
{
    die('Error: ' . mysqli_error($con));
}
echo "wait for conformation";

mysqli_close($con);
?>

```

4.1.2.2 Code For Student login check

```

<?php
    //Start session
    session_start();
    //Unset the variables stored in session
    unset($_SESSION['SESS_username']);
    unset($_SESSION['SESS_password']);
?>
<!DOCTYPE html>
<html xml:lang="EN" lang="EN" dir="ltr">
<head>
<title>SOE LIBRARY INFORMATION SYSTEM</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
<meta http-equiv="imagetoolbar" content="no" />
<link rel="stylesheet" href="../styles/layout.css" type="text/css" />
<!-- Homepage Specific Elements -->
<script type="text/javascript" src="../scripts/jquery-1.4.1.min.js"></script>
<script type="text/javascript" src="../scripts/jquery-ui-1.7.2.custom.min.js"></script>
<script type="text/javascript" src="../scripts/jquery.tabs.setup.js"></script>
<!-- End Homepage Specific Elements -->
</head>
<body id="top">
<div class="wrapper row1">
    <div id="header" class="clear">
        <div class="fl_left">

```



```

        <h1><a href="homepage.php">SOE LIBRARY MANAGEMENT
SYSTEM</a></h1>
        <p></p>
    </div>
    <div class="fl_right">
        <ul>
            <li><a
href="http://localhost/xampp/project/library/homepage.php">Home</a></li>
            <li><a
href="http://localhost/xampp/project/library/contact/Contactus.php">Contact
Us</a></li>
            <li><a
href="http://localhost/xampp/project/library/egateway/egateway.php">E-
Gateway</a></li>
            <li><a
href="http://localhost/xampp/project/library/studentlogin/studentlogin.php">Student
Login</a></li>
            <li><a href="teacherlogin.php">Teacherlogin</a></li>
            <li><a
href="http://localhost/xampp/project/library/admin/adminlogin.php">Adminlogin</a
></li>
        </ul>
        <form action="#" method="post" id="sitesearch">
            <fieldset>
                <strong>Search:</strong>
                <input type="text" value="Search Our webiste for books;"
onfocus="this.value=(this.value=='Search Our website for books;')? " : this.value ;"
/>
                <input type="image" src="../images/search.gif" id="search" alt="Search" />
            </fieldset>
        </form>
    </div>
</div>
<!--
#####
##### -->
<div class="wrapper row2">
    <div class="rnd">
        <!-- ##### -->
        <div id="topnav">
            <ul>
                <li class="active"><a
href="http://localhost/xampp/project/library/homepage.php">Home</a></li>
                <li><a href="http://localhost/xampp/project/library/newarrivals.php">New
Arrivals</a></li>

```

```

</div>
</div>
</body>
</html>

<?php
    session_start();
    $host="localhost"; // Host name
    $username="root"; // Mysql username
    $password=""; // Mysql password
    $db_name="library"; // Database name
    $tbl_name="studentlogin"; // Table name

    // Connect to server and select database.
    mysql_connect("$host", "$username", "$password")or die("cannot connect");
    mysql_select_db("$db_name")or die("cannot select DB");

    // username and password sent from form
    $myusername=$_POST['username'];
    $mypassword=$_POST['password'];
    $sql="SELECT * FROM $tbl_name WHERE username='$myusername' and
password='$mypassword'";
    $result=mysql_query($sql);

    // Mysql_num_row is counting table row
    $count=mysql_num_rows($result);

    // If result matched $myusername and $mypassword, table row must be 1 row

    if($count>0)
    {
        session_regenerate_id();
        $member = mysql_fetch_assoc($result);
        $_SESSION['SESS_username'] = $member['username'];
        $_SESSION['SESS_password'] = $member['password'];
        session_write_close();
        header("location:studentlogin1.php");
        exit();
    }else {
        //Login failed
        $errmsg_arr[] = 'user name and password not found';
        $errflag = true;
        if($errflag) {
            $_SESSION['ERRMSG_ARR'] = $errmsg_arr;
            session_write_close();
            header("location: studentlogin.php");
            exit();
        }
    }
}

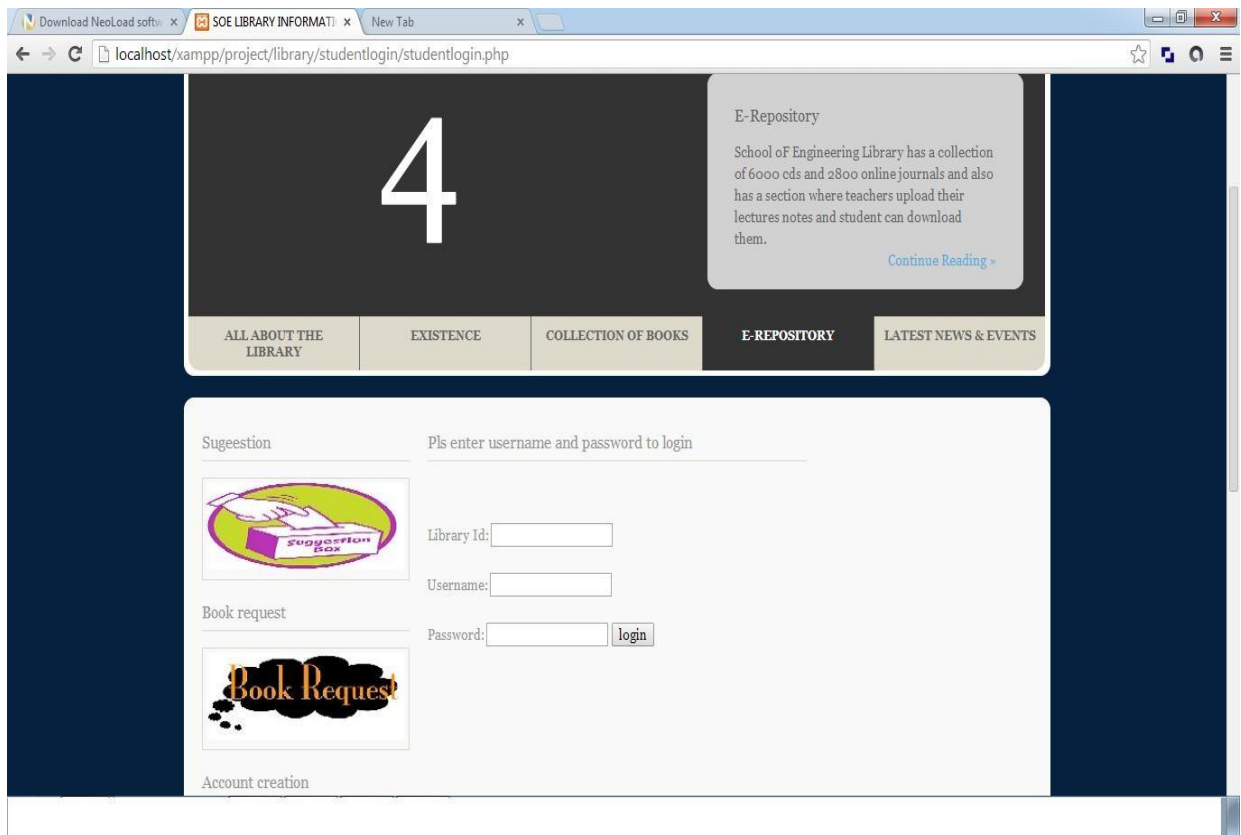
```

```
}  
}  
?>
```

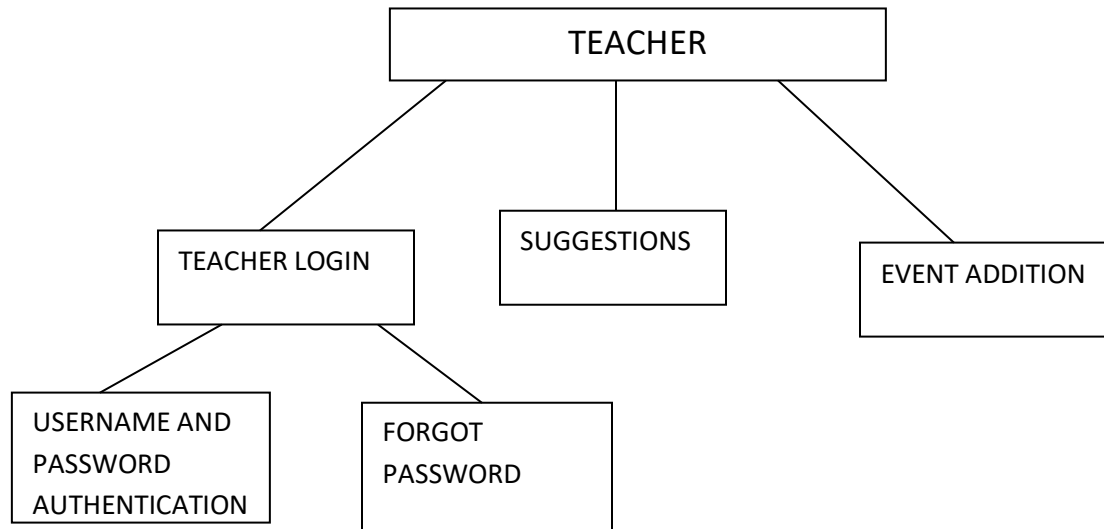
4.1.2.3 Code For Event Addition

```
<?php  
$con=mysqli_connect("localhost","root","","admin");  
// Check connection  
if (mysqli_connect_errno())  
{  
    echo "Failed to connect to MySQL: " . mysqli_connect_error();  
}  
  
$sql="INSERT INTO event(Name,date,time,venue,mname,contactno,email)  
VALUES  
('$ _POST[ename]','$ _POST[date]','$ _POST[time]','$ _POST[venue]','$ _POST[name  
]','$ _POST[phno]','$ _POST[email]')";  
if (!mysqli_query($con,$sql))  
{  
    die('Error: ' . mysqli_error($con));  
}  
    echo "wait for conformation";  
    mysqli_close($con);  
?>
```

4.1.2 Screenshot for Student login



- **4.1.3 Teacher Module**



The following module contains various facilities like teacher login, suggestions, and event addition. Further any teacher if at any moment forgets his/her password he/she can retrieve it from ‘forgot password’ option.

4.1.3.1 Code For Teacher Login

```
<?php
    session_start();
    $host="localhost"; // Host name
    $username="root"; // Mysql username
    $password=""; // Mysql password
    $db_name="library"; // Database name
    $tbl_name="teacherlogin"; // Table name

    // Connect to server and select databse.
    mysql_connect("$host", "$username", "$password")or die("cannot connect");
    mysql_select_db("$db_name")or die("cannot select DB");

    // username and password sent from form
    $myusername=$_POST['username'];
    $mypassword=$_POST['password'];
    $sql="SELECT * FROM $tbl_name WHERE username='$myusername' and
password='$mypassword'";
```

```

$result=mysql_query($sql);

// Mysql_num_row is counting table row
$count=mysql_num_rows($result);

// If result matched $myusername and $mypassword, table row must be 1 row

if($count>0)
{
    session_regenerate_id();
    $member = mysql_fetch_assoc($result);
    $_SESSION['SESS_username'] = $member['username'];
    $_SESSION['SESS_password'] = $member['password'];
    session_write_close();
    header("location:teacherlogin1.php");
    exit();
}else {
    //Login failed
    $errmsg_arr[] = 'user name and password not found';
    $errflag = true;
    if($errflag) {
        $_SESSION['ERRMSG_ARR'] = $errmsg_arr;
        session_write_close();
        header("location: teacherlogin.php");
        exit();
    }
}
?>

```

4.1.3.2 Code for Suggestion

```

<?php
session_start();
?>
<!DOCTYPE html>
<html xml:lang="EN" lang="EN" dir="ltr">
<head>
<title>SOE LIBRARY INFORMATION SYSTEM</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
<meta http-equiv="imagetoolbar" content="no" />
<link rel="stylesheet" href="../styles/layout.css" type="text/css" />
<!-- Homepage Specific Elements -->
<script type="text/javascript" src="../scripts/jquery-1.4.1.min.js"></script>
<script type="text/javascript" src="../scripts/jquery-ui-
1.7.2.custom.min.js"></script>

```

<p>School oF Engineering library came into existence in year 1995 and from then on it has been a premier source of information for college students</p>

<p class="readmore">Continue Reading »</p>

</div>

</div>

<div class="featured_box" id="fc3">

<div class="floater">

<h2>Collection</h2>

<p>School of Engineering Library has a collection of over 25000 books related to various subjects in engineering</p>

<p class="readmore">Continue Reading »</p>

</div>

</div>

<div class="featured_box" id="fc4">

<div class="floater">

<h2>E-Repository</h2>

<p>School oF Engineering Library has a collection of 6000 cds and 2800 online journals and also has a section where teachers upload their lectures notes and student can download them.</p>

<p class="readmore">Continue Reading »</p>

</div>

</div>

<div class="featured_box" id="fc5">

<div class="floater">

<h2>Latest News and events</h2>

<p></p>

<p class="readmore">Click here to go to latest news page;</p>

</div>

</div>

</div>

<ul id="featured_tabs">

All About The Library

Existence

Collection of Books

E-Repository

<li class="last">Latest News & Events

<div class="overlay_right"></div>

<!-- ##### -->

</div>

</div>

```

        $section= $row['section'] ;
        $yearofadm= $row['yearofadm'] ;
    }

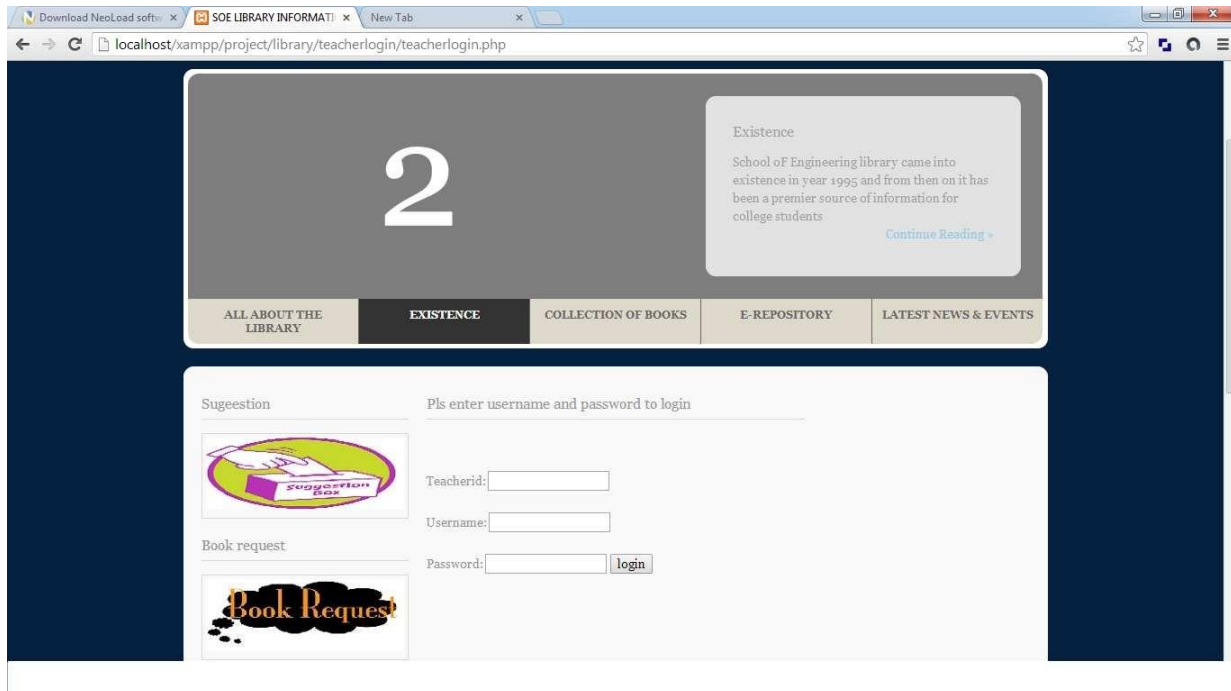
}
mysqli_close($connection);
?>
<?php
    require'../include/connection.php';

$sql="INSERT INTO suggest ( libid,suggestion)
VALUES
('$libid','$_POST[suggestion])";

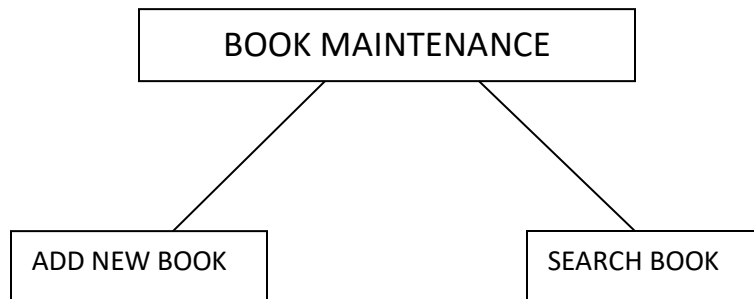
if (!mysqli_query($con,$sql))
{
    echo "error";
}
echo "successs";
mysqli_close($con);
?>

```

4.1.3 Screenshot for teacher's login



- **4.1.4 Book Module**



The following module contains various facilities like add new book and search book. In the 'add new book' section if any new book comes in the library then the librarian can add its specifications. Similarly if the user wants to search for a specific book then he/she can use search book option to do it.

4.1.4.1 Code For Adding New books

```
<?php
$con=mysqli_connect("localhost","root","","library");
// Check connection
if (mysqli_connect_errno())
{
    echo "Failed to connect to MySQL: " . mysqli_connect_error();
}

$sql="INSERT INTO books (code, bookname, author, publication, subject,
numberofbooks)
VALUES
('$_POST[bookid]','$_POST[bookname]','$_POST[author]','$_POST[publication]','$_P
OST[subject]','$_POST[numberofbooks])";

if (!mysqli_query($con,$sql))
{
    echo "error";
}
echo "successs";
```

```

mysqli_close($con);
?>
<?php
$con=mysqli_connect("localhost","root","","admin");
// Check connection
if (mysqli_connect_errno())
{
    echo "Failed to connect to MySQL: " . mysqli_connect_error();
}

$sql="INSERT INTO newarrivals (code, bookname, author, publication, subject,
numberofbooks,arrivaldate)
VALUES
('$ _POST[bookid]','$ _POST[bookname]','$ _POST[author]','$ _POST[publication]','$ _P
OST[subject]','$ _POST[numberofbooks]','$ _POST[arrivaldate]')";

if (!mysqli_query($con,$sql))
{
    echo "error";
}
echo "successs";
mysqli_close($con);
?>

```

4.1.4.2 Code For Book Search

```

<!DOCTYPE html>
<html xml:lang="EN" lang="EN" dir="ltr">
<head>
<title>SOE LIBRARY INFORMATION SYSTEM</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
<meta http-equiv="imagetoolbar" content="no" />
<link rel="stylesheet" href="../styles/layout.css" type="text/css" />
<!-- Homepage Specific Elements -->
<script type="text/javascript" src="../scripts/jquery-1.4.1.min.js"></script>
<script type="text/javascript" src="../scripts/jquery-ui-1.7.2.custom.min.js"></script>
<script type="text/javascript" src="../scripts/jquery.tabs.setup.js"></script>
<!-- End Homepage Specific Elements -->
</head>
<body id="top">
<div class="wrapper row1">
    <div id="header" class="clear">
        <div class="fl_left">
            <h1><a href="homepage.php">SOE LIBRARY MANAGEMENT
SYSTEM</a></h1>

```

```

<div id="left_column">
  <h2>Sugeestion</h2>
  <div class="imgholder"><a href="#"></a></div>
  <h2>Book request</h2>
  <div class="imgholder"><a href="#"></a></div>
  <h2>Account creation</h2>
  <div class="imgholder"><a
href="http://localhost/xampp/project/library/create/createaccount.php"></a></div>
  <h2>Photos</h2>
  <div class="imgholder"><a href="#"></a></div>
</div>
<!-- ##### -->
<div id="latestnews">
  <h2>ENTER BOOK DETAILS</h2>

```

```

    <table>
    <form action="advancesearchresult.php" method="POST">
    <legend>Book Information:</legend>
    <tr>
    <td>Book Code:</td> <td><input type="text" name="bookcode"
size="30"><br><br></td>
    </tr>
    <tr>
    <td>Book Name: </td> <td><input type="text" name="bookname"
size="30"><br><br></td>
    </tr>
    <tr>
    <td>Publication: </td> <td> <input type="text" name="publication"
size="30"><br><br></td>
    </tr>
    <tr>
    <td>Author Name: </td> <td> <input type="text" name="authername"
size="30"><br><br></td>
    </tr>
    <tr>
    <td><p>Click the submit</p></td>
    </tr>
    <tr>
    <td height="40"><input type="submit" name="submit" value="submit" action=
"insertaccount.php"></td>
    </tr>

```

```

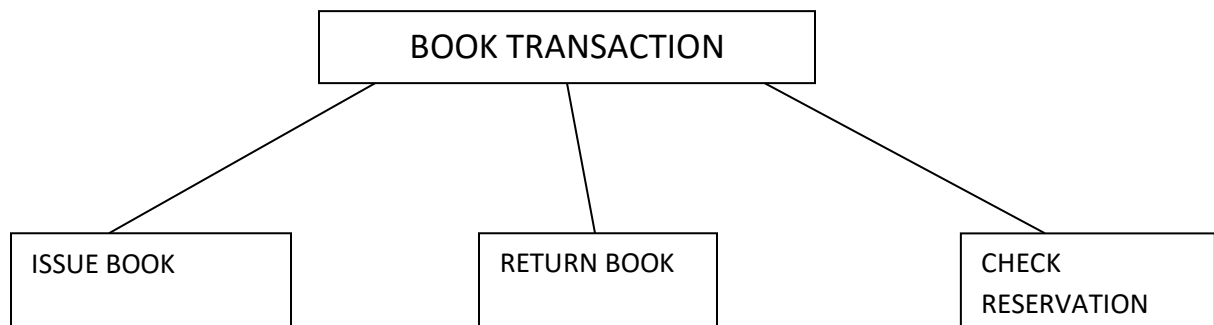
</tr>";

while($row = mysqli_fetch_array($result))
{
    echo "<tr>";
    echo "<td>" . $row['code'] . "</td>";
    echo "<td>" . $row['bookname'] . "</td>";
    echo "<td>" . $row['author'] . "</td>";
    echo "<td>" . $row['publication'] . "</td>";
    echo "<td>" . $row['subject'] . "</td>";
    echo "<td>" . $row['numberofbooks'] . "</td>";
    echo "</tr>";
}
echo "</table>";

mysqli_close($con);
?>

```

- **4.1.5 Book Transaction Module**



The following module contains various facilities like issue book, return book and check reservation.

4.1.5.1 Code For issue book

```
<?php
require'../include/connection1.php';
$result = mysqli_query($connection,"SELECT * FROM studentlogin");
while($row = mysqli_fetch_array($result))
{
    $libid=$row['logid'] ;
    $username= $row['username'];
    $password= $row['password'];
    $numbooks=$row['numbooks'];
}
    if($numbooks>3)
    {
        echo"cannot issue books limit exceeded";
    }
    else
    {
        $libid=$_POST['libid'];
        $bookid=$_POST['bookid'];
        $issuedate=$_POST['issuedate'];

        $returndate=date_add($issuedate,date_interval_create_from_date_string("15 days"));
        $sql="INSERT INTO issuebooks(libid,bookid,issuedate,returndate)
        VALUES
        ($libid,'$bookid','$issuedate','$returndate)";
        mysqli_query($connection,"UPDATE studentlogin SET
numbooks=numbooks+1
        WHERE logid='$libid'");
        mysqli_query($connection,"UPDATE book SET
numberofbooks=numberofbooks-1
        WHERE code='$bookid'");
        if (!mysqli_query($connection,$sql))
        {
            die('Error1: ' . mysqli_error($connection));
        }
        else
        echo"added 1 record";
    }
    mysqli_close($connection);
?>
```

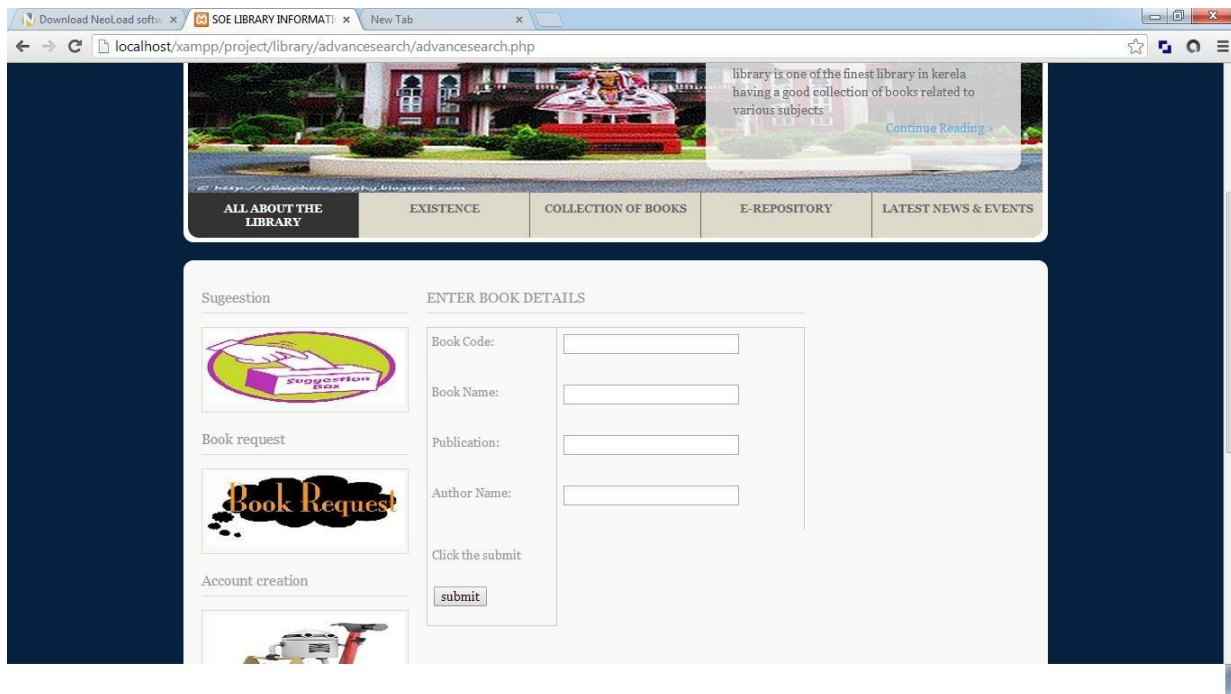
4.1.5.2 Code For book Return

```

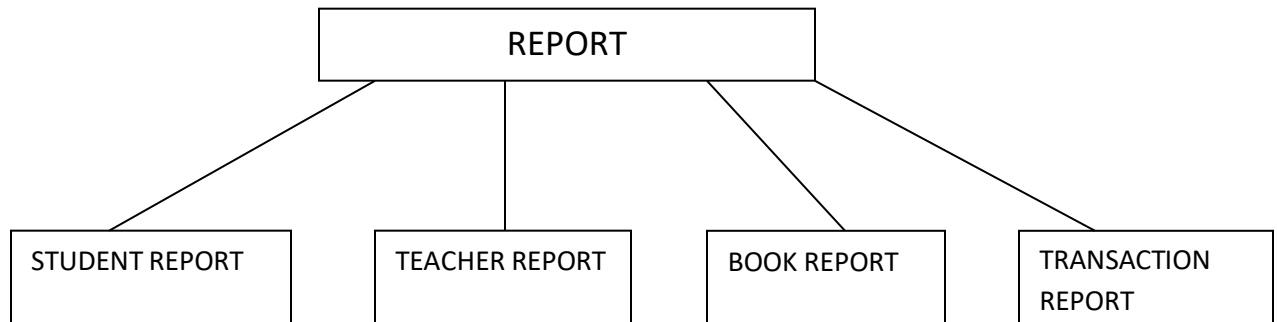
<?php
require'../include/connection1.php';
$libid=$_POST['libid'];
$bookid=$_POST['bookid'];
$result = mysqli_query($connection,"SELECT * FROM issuebooks where libid= '$libid' &&
bookid='$bookid'");
    $rowcount=mysqli_num_rows($result);
    if($rowcount>0)
    {
        $sql=mysqli_query($connection,"UPDATE studentlogin SET
numbooks=numbooks-1
        WHERE logid='$libid'");
        $sql1=mysqli_query($connection,"UPDATE book SET
numberofbooks=numberofbooks+1
        WHERE code='$bookid'");
        $sql2=mysqli_query($connection,"DELETE FROM issuebooks WHERE
libid=$libid && bookid='$bookid'");
    }
    else
        echo"no book issued";
    mysqli_close($connection);
?>

```

4.1.2 Screenshot for Book search



- **4.1.6 Report Module**



The following module contains various facilities like student report, teacher report, book report, and transaction report.

Code For Report Generation

4.1.6.1 Code For Teacher report

```
<?php
require '../include/connection1.php';
$result = mysqli_query($connection,"SELECT * FROM
teacher");
$count=mysqli_num_rows($result);
echo "<table border='1' style='width=100px'>
<tr>
<td>Teacher id</td>
<td>Name</td>
<td>Designation</td>
<td>Branch</td>
<td>Contact no</td>
</tr>";
while($count>0)
{
while($row = mysqli_fetch_array($result))
```

```

        {echo "<tr>";
        echo "<td>" . $row['tid'] . "</td>";
        echo "<td>" . $row['name'] . "</td>";
        echo "<td>" . $row['designation'] . "</td>";
        echo "<td>" . $row['branch'] . "</td>";
        echo "<td>" . $row['contactno'] . "</td>";
        echo "</tr>";
        }
        $count=$count-1;
    }

    mysqli_close($connection);
?>
</table>

```

4.1.6.2 Code For STUDENT REPORT

```

<?php
    require '../include/connection1.php';
    $result = mysqli_query($connection,"SELECT * FROM
student");
    $count=mysqli_num_rows($result);
    echo "<table border='1' style='width=100px'>
    <tr>
    <td>Libraryid</td>
    <td>Registration number</td>
    <td>Name</td>
    <td>Branch</td>
    <td>Semester</td>
    <td>Section</td>
    <td>Year of adm</td>
    </tr>";
    while($count>0)
    {
        while($row = mysqli_fetch_array($result))
        {echo "<tr>";
        echo "<td>" . $row['libid'] . "</td>";
        echo "<td>" . $row['regno'] . "</td>";
        echo "<td>" . $row['stuname'] . "</td>";
        echo "<td>" . $row['branch'] . "</td>";
        echo "<td>" . $row['semester'] . "</td>";
        echo "<td>" . $row['section'] . "</td>";
        echo "<td>" . $row['yearofadm'] . "</td>";

```



```

echo"</tr>";
}
$count=$count-1;
}

mysqli_close($connection);
?>
</table>

```

4.1.6.3 Code For Issue Report

```

<?php
require '../include/connection1.php';
$result = mysqli_query($connection,"SELECT * FROM
issuebooks order by libid");
$count=mysqli_num_rows($result);
echo "<table border='1' style='width=100px'>
<tr>
<td>Libraryid</td>
<td>Bookid</td>
<td>Issue date</td>
<td>Return date</td>
</tr>";
while($count>0)
{
while($row = mysqli_fetch_array($result))
{echo "<tr>";
echo "<td>" . $row['libid'] . "</td>";
echo "<td>" . $row['bookid'] . "</td>";
echo "<td>" . $row['issuedate'] . "</td>";
echo "<td>" . $row['returndate'] . "</td>";
echo"</tr>";
}
$count=$count-1;
}

mysqli_close($connection);
?>
</table>

```

4.1.6.4 Code For Book Report

```
<?php
require '../include/connection1.php';

$result = mysqli_query($connection,"SELECT * FROM books")
$count=mysqli_num_rows($result);
echo "<table border='1' style='width=100px'>
<tr>
<td>Book Id</td>
<td>Name</td>
<td>Publication</td>
<td>Author</td>
<td>Subject</td>
<td>Number Of Books</td>
</tr>";
while($count>0)
{
while($row = mysqli_fetch_array($result))
{echo "<tr>";
echo "<td>" . $row['code'] . "</td>";
echo "<td>" . $row['bookname'] . "</td>";
echo "<td>" . $row['publication'] . "</td>";
echo "<td>" . $row['author'] . "</td>";
echo "<td>" . $row['subject'] . "</td>";
echo "<td>" . $row['numberofbooks'] . "</td>";
echo "</tr>";
}
$count=$count-1;
}
mysqli_close($connection);
?></table>
```

CHAPTER 5

SYSTEM TESTING

The aim of the system testing process was to determine all defects in our project .The program 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.

Our Project went through two levels of testing

- 1.Unit testing
- 2.integration testing

UNIT TESTING

Unit testing is undertaken when a module has been created and succesfully reviewed .In order to test a single module we need to provide a complete environment ie 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
- 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 .

2. Test for Student login module

- Test for Student login Form-This form is used for log in of Student .In this we enter thelibraryid, username and password if all these are correct student login page will open other wise 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 logg in of teacher .In this we enter the username and password if all these are correct teacher login 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.

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 6

CONCLUSION & FUTURE SCOPE

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 and also give necessary suggestion to library and also 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

CHAPTER 7

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
- <https://www.javatpoint.com/phpmyadmin>
- <https://app.creately.com/d/IIB3y7rSN7n/edit>
- <https://ijiet.com/wpcontent/uploads/2014/12/4.pdf>
- <https://ijisrt.com/wp-mysqldatabase>
- Fundamentals of software engineering by Rajib mall, PHIlearning
- Web development and application development by Ivan Byross BPB publications