Ridwan Hossan

Designing a Library Management System.

Metropolia University of Applied Sciences

Bachelor of Engineering

Information technology

Bachelor's Thesis

1st May 2022

Abstract

Author:	Ridwan Hossan
Title:	Designing a Library Management System
Number of Pages:	31 pages
Date:	1st May 2022
Degree:	Bachelor of Engineering
Degree Programme:	Information Technology
Professional Major:	Software Engineering
Supervisors:	Janne Salonen, Head of School, School of ICT

Traditional Libraries can be turned into digital libraries using the Library Management System (LMS). LMS allows a user to maintain a library efficiently and keep records of the books and customers accurately. The biggest advantage of LMS is it saves time and money and needs less human interaction. The purpose of the thesis is to develop a LMS system, which can perform basic tasks of a library.

To develop the project the author used JavaScript and Bootstrap framework for user interface design and for the Backend part the author used PHP and MySQL database. For initial planning and to visualize the project the author used UML diagram methodologies. After strategizing the project with a use case diagram, sequence diagram, activity diagram and ER diagram the author developed the Frontend and Backend part of the project.

In the end, the author tested the basic functionality of the project. The testing results were successful and fulfilled the author's expectations. The Library Management System is still in the prototype stage and needs further development for actual use.

Keywords: MySQL, PHP, Library Management System

Contents

List of Abbreviations

GUI:	Graphical user interface.
DMS:	Database management system.
LMS:	library management system.
HTML:	HyperText Markup Language.
CSS:	Cascading style sheets.
RDMS:	Relational database management system.
UML:	Unified modeling language.

1 Introduction

A library is a collection of well-organized information and resources that is made available for borrowing to a certain group of people. With the advent of technology, it is very important to make all systems more user-friendly. Library Management System (LMS) is a software program that allows physical libraries to be converted into digital libraries. A Library Management System (LMS) allows the user to keep track of both the books and the customers. The system's deployment in the company will significantly decrease data entry, time, and deliver easily computed results. It assists in keeping track of all information on the books in the library, including their pricing, status, and total quantity. Instead of utilizing the manual writing system, the user will find it easier to use this automated system. [2]

The main purpose of the project is to is to build an online LMS system by using different software methodologies. The author designed the LMS with the basic and necessary features like the admin/librarian can add/delete book, can add/delete author, can issue books, can manage books and users, and users/students can search and borrow books. For Frontend part of the project the author used HTML, CSS, JavaScript and Bootstrap framework and for the Backend part the author used PHP and MySQL database. For Unified modelling language (UML) design the author used Lucidchart.

The author divided the thesis into three sections. In first section the author planed the project through UML methodologies. In the second section the author designed the graphical user interface of the project. And in the third section the author presented some of the code snippet of the project and tests the outcome.

2 Tools used for the project

The author used HTML, CSS, JavaScript and Bootstrap framework for Frontend part. Frontend development is a type of computer programming that focuses on

the coding and building of a website, which will be visible to the user. It's about ensuring that a website's visual elements are functioning. And for the Backend part the author uses PHP and MySQL. The server side of a program and anything that connects between the database and the browser is referred to as back-end development.

2.1 PHP

PHP is a scripting language developed by Rasmus Lerdorf in 1994. Originally, he wrote several C programs to execute various tasks on his static website, which he had constructed for himself on the internet. He had unknowingly established the groundwork for a new programming language. The primary goal was to connect with the servers and databases where his static files were stored.
PHP is a server-side programming language that is free, open sourced and object oriented. PHP is a good option for developing websites. [7]
PHP is a server-side scripting language that is used to create interactive web applications that are linked to a MySQL database. It is responsible for the website's dynamic content, database, and session tracking functions. It has the ability to access the cookies variable as well as set cookies.
It contributes to the encryption of data and the application of validation.
PHP is compatible with a variety of protocols, including HTTP, POP3, SNMP and may more. One can manage which pages of the website a user can view by using the PHP programming language. The fact that PHP is simple to install and configure is one of the primary reasons why PHP is the finest language to learn.
PHP is capable of handling forms in a variety of ways, including collecting data from users via forms, storing it in a database, and returning valuable information to the user. [7]

2.2 SQL

In computing, SQL stands for Structured Query Language, and it is a programming language that may be used to store, alter, and retrieve data from a database system. A relational database, or SQL, is another term for this kind of database. Based on set theory and predicate logic, the relational model makes data retrieval easier, assures data integrity and provides a database structure that is not reliant on the applications that access the data contained in the database, among other benefits. The relational paradigm is built on the concept of connection. A relation is a table-like structure that has a set of columns and rows that describe a single entity made up of connected data. [8, p, 5]

The reason why SQL is widely used are-

- A relational database management system (RDBMS) allows users to obtain access to data.
- Users may make advantage of this functionality to provide explanations for the data.
- Provides users with the ability to specify and govern the data stored in a database.
- Enables the embedding of SQL modules, libraries, and pre-compilers into other programming languages.
- Users have the power to create databases and tables, as well as remove them.
- Provides the ability for users to create database views, stored procedures, and functions.[1]

There are 3 types of SQL: Data definition language, Data control language and data manipulation language. Data definition language is used to create and manipulate tables, views, schemas, domains, triggers, and stored procedures. The keywords are CREAT, ALTER and DROP. The data control language controls user access to object. The DCL keywords are GRANT and REVOKE. Data manipulation language is used to modify or remove database objects. The Data manipulation language keywords are SELECT, INSERT, UPDATE and DELETE. [1]

2.2.1 MySQL

MySQL is a SQL-based relational database management system. MySQL enables users to manage, save, change, and remove data, as well as organize data. MySQL has an integrated tool called MySQL Workbench that simplifies the process of developing, designing, and constructing databases. MySQL comes in a variety of forms and is updated often. MySQL, being an open-source platform, provides a large and active community of users and developers. [10]

2.3 Bootstrap

Bootstrap is a framework that works with HTML, CSS, and JavaScript to create dynamic and device-friendly webpages. It's a simple and quick web development framework to use. It's a framework for front-end development. For websites and applications, developers may create multifunctional, user-friendly, and beautiful front-end designs.

Bootstrap comes with ready-to-use components that make it easier for developers to design websites on a tight schedule. Developers don't have to start from the beginning, and they may tweak particular components depending on one’s suggestions to make it unique. Whether an expert or a novice in web design, a user may effortlessly install Bootstrap. Users do not need to be experts in internet technology (HTML, CSS, or JavaScript) to get started; they may start with the basics. [4]

2.4 JavaScript

JavaScript is a scripting language Which makes a website more responsive and lucrative. Brendan Eich came up with the idea of JavaScript (co-founder of the Mozilla project, the Mozilla Foundation, and the Mozilla Corporation). JavaScript is a scripting language for beginners with a wide variety of features. As one gets more experience with JavaScript, one will be able to create games, animated 2D and 3D images, database-driven systems, and more. [6]

JavaScript is a programming language that is both tiny and powerful. On top of the core JavaScript programming language, developers have built a variety of tools that allow users to get access to a significant amount of functionality with little effort. Here are only a few examples:

Among the functions provided by web browser APIs include the ability to dynamically compose HTML and CSS styles, gather and analyse video feeds from a user's camera, and generate 3D pictures and audio samples.
APIs from third-party content providers, such as Twitter or Facebook, enable developers to incorporate elements from their own websites into those of other content producers. Third-party frameworks and libraries that can be used in conjunction with HTML to accelerate the construction of website and mobile app. [6]

3 Unified Modelling Language (UML)

It is abbreviated as UML, which stands for Unified Modeling Language. It is a standardized modeling language consisting of an integrated set of diagrams that was developed to assist system and software developers in specifying, visualizing, constructing, and documenting the artifacts of software systems, as well as for business modeling and other non-software systems. With the Unified Modeling Language (UML), users may model big and complex systems using best engineering techniques that have been shown to work in previous modeling

projects. The Unified Modeling Language (UML) is a critical component in developing object-oriented applications and the software development life cycle. The Unified Modeling Language (UML) is a graphical notation system that is used to represent the design of software projects. When project teams use the UML, they may communicate more effectively, explore possible designs, and verify the architectural design of the program. [5]

To design the UML diagram the author used Lucidchart. Lucidchart is a web-based diagramming program that enables users to graphically collaborate on designing, modifying, and sharing charts and diagrams. It is available for both Mac and Windows users.

3.1 Use case diagram

A use-case model describes a system's functional requirements in terms of use cases. It is a model of the system's intended functionality (use cases) and its environment (actors). It is possible to describe one’s system's users (sometimes referred to as actors in UML) and their interactions with the system using a use case diagram. Use cases is build using a set of specific symbols and relationships. [5]

User stories-

Authors Online library Management System divided in two modules User and Admin

Admin Features Consists of

- By login admin can view own dashboard.
- Can add/edit/ erase category
- Can add/edit/ erase author.

- Can add/edit/ erase books.
- Can issue a new book to user and edit the details after returning the book.
- Can search user by using their user ID.
- Can also view user details.
- Can change own password.

User features

- Can sign up themselves and get user id.
- By login user can view own dashboard.
- User can view all the books in the library.
- Can edit own profile and password.
- Can view issued book and book return date.

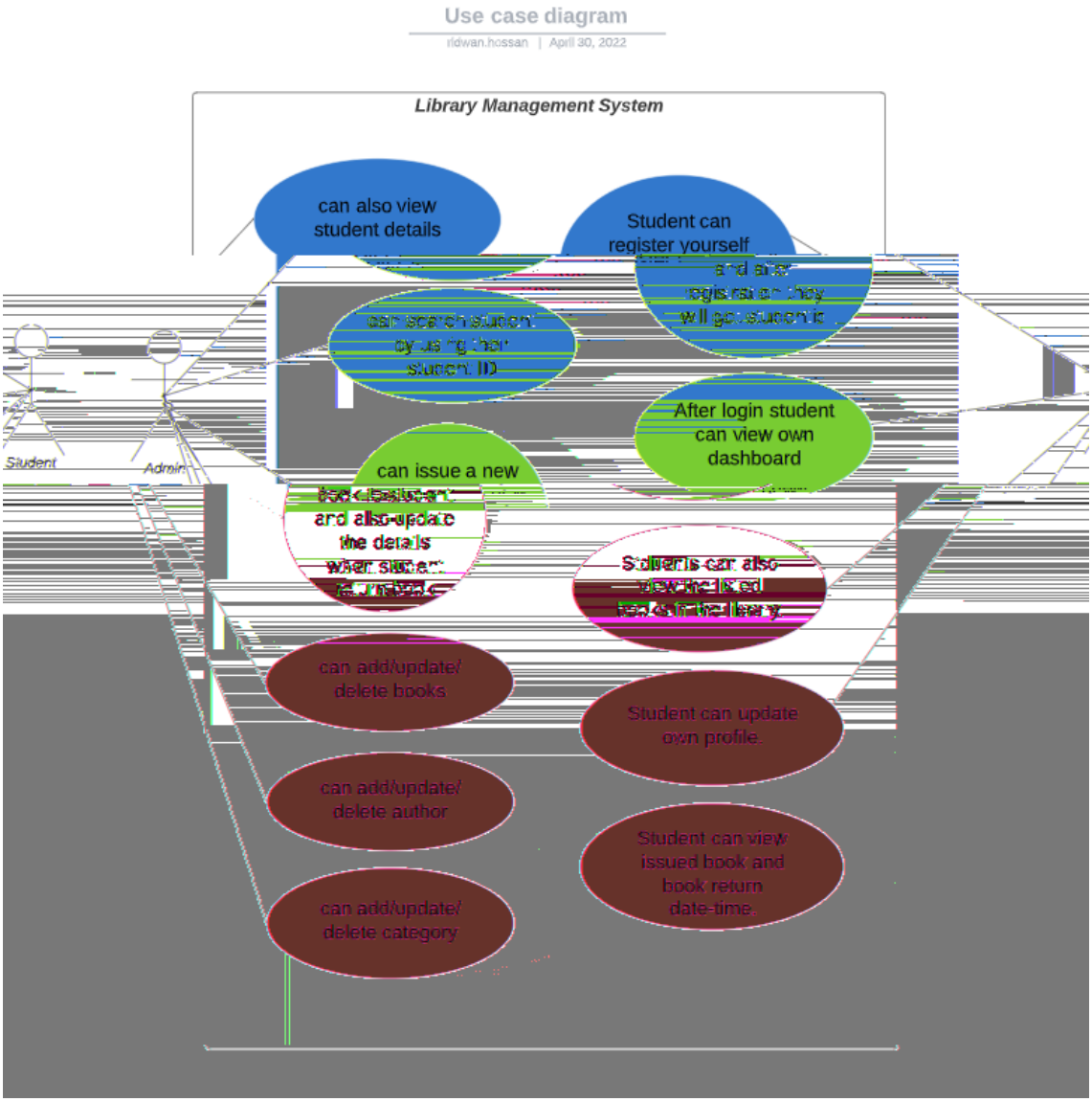


Figure 1: Use case diagram

3.2 ER diagram

It is necessary to develop an entity-relationship diagram (ERD) in order to establish a good database design. In database design, high-level analytical data model that may be used to develop the conceptual architecture of the database. An entity is a real-world item or concept that exists independently of any other thing or idea. Entities in a relational database are analogous to database tables, with each row of the table represents a single instance of the item in the database. An entity's attribute is a specific value that determines the entity. A relationship is a connection that defines how two or more entities interact. Cardinality is the number of occurrences of one entity that can or must relate to every occurrence of another entity in the context of ERD. Relationships can be one-to-one, one-to-many, or many-to-many in general. [5]

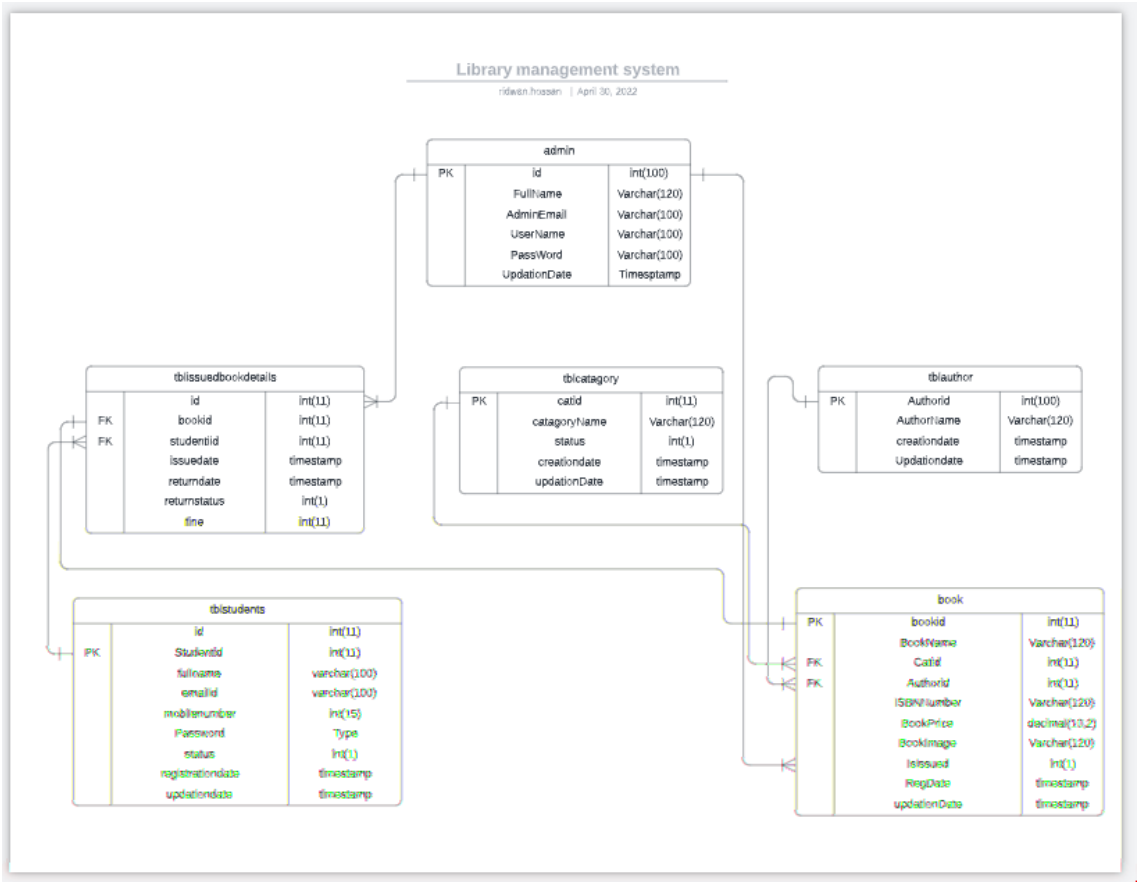


Figure 2: ER Diagram

The authors library database consists of 6 tables,

- Admin- In this table the admin details is stored. Admin can access all the tables.
- tblStudent- All the information of the users are saved.
- Book- All the information of the books are saved.
- tblAuthor- All the information of the authors are stored.
- tblCatagory- All the category of books are saved.
- tblissuebook- Information's of all issued book to the users are saved.

The relationship among the tables is such that a student can be issued one to many books, but a book can be borrowed by only one student at a time. A book can have one to many authors and an author can write one to many books. A book can have one category, but one category can have many books. Admin can add/edit/delete books, admin can issue books, admin view student details and add/edit/delete category and author.

After implementing these tables, the visual output of the database is-

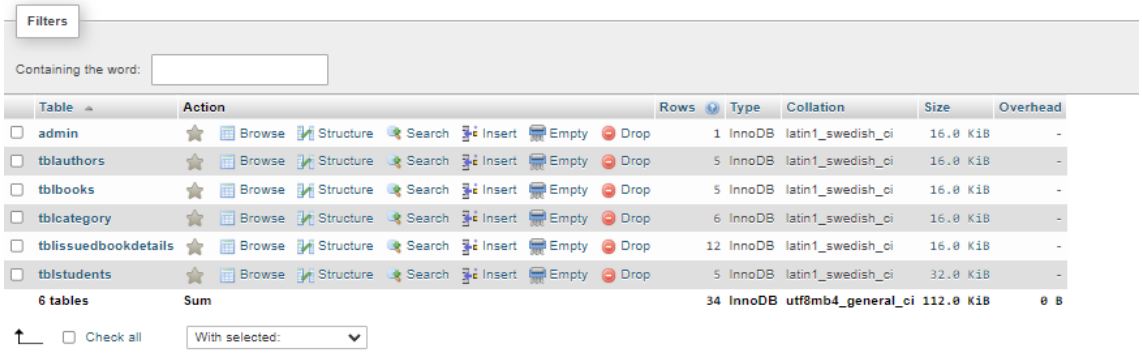


Figure 3: Tables of the database

3.3 Sequence diagram

Sequence Diagrams are used to show object cooperation by depicting them as a time sequence. A use case scenario is a representation of how things interact with one another in a specific use case scenario. With the strong visual modeling capability, it is possible to create sophisticated sequence diagrams in a matter of seconds. A sequence diagram may also be generated by specific modeling tools, such as Visual Paradigm, on the basis of the sequence of events described in the use case situation. [5]

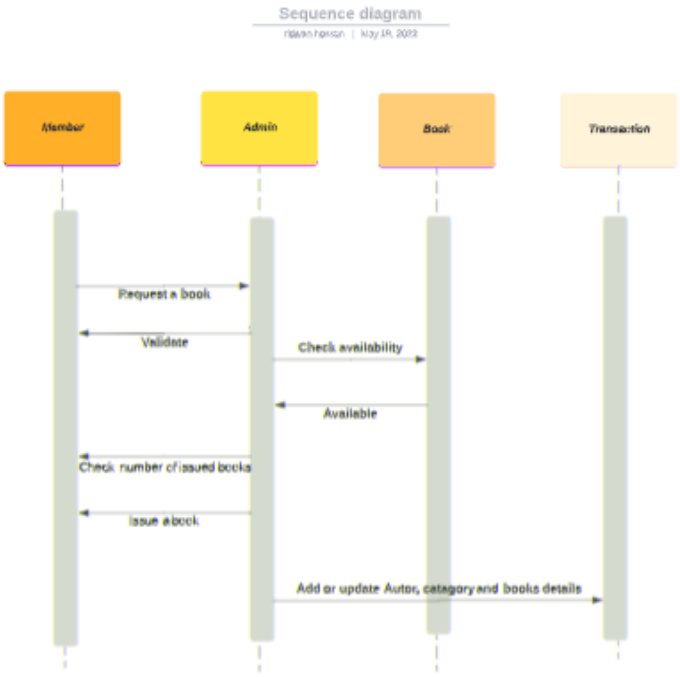


Figure 4: Sequence diagram

According to the sequence diagram A student request a book and the admin validate the student id. Next the admin checks the availability of the book and after founding it available the admin checks the number of issued book of the student and check any overdue books than the admin issues a book to the student. Admin can add and update author, category, and book details.

3.4 Activity diagram

Activity diagrams are graphical presentation of processes that include choice, iteration, and concurrency. It specifies the target system's control flow, such as

complicated business rules and processes, as well as the use case and business process. Activity diagrams are used to represent both computing and organizational processes in the Unified Modelling Language (UML) (i.e., workflows). [5]

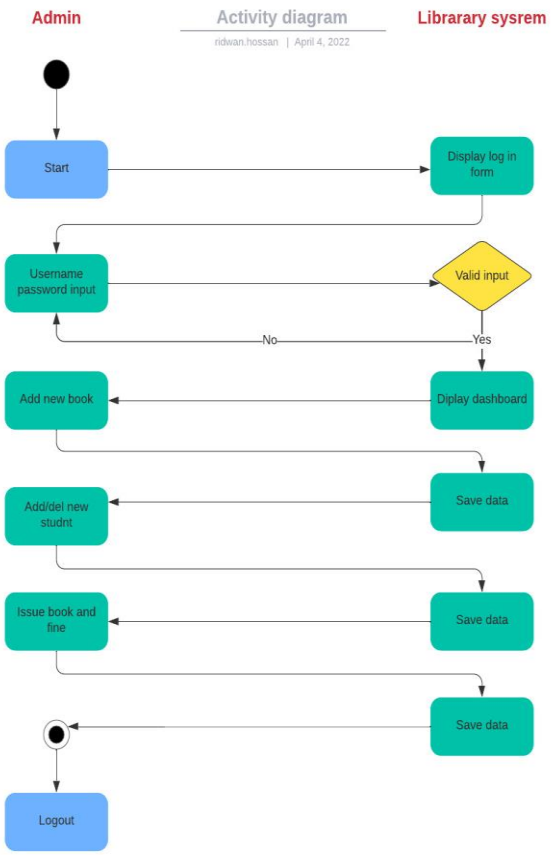


Figure 5: Admin activity diagram

According to the admin activity diagram when the admin enters to the library management webpage it displays the login form, where the admin has to put the username and password and click enter. If the user credentials match to the database, it displays the admin dashboard and if admin credentials doesn't match the system display again the login page. From the admin dashboard the admin

can add or delete a book and save the data in the database. The admin can view student details and put them inactive mode if they have overdue books. The admin can issue books to the student and also can issue a fine if the student doesn't return there borrow books on time. After finishing all the transections, the admin can log out from the system.

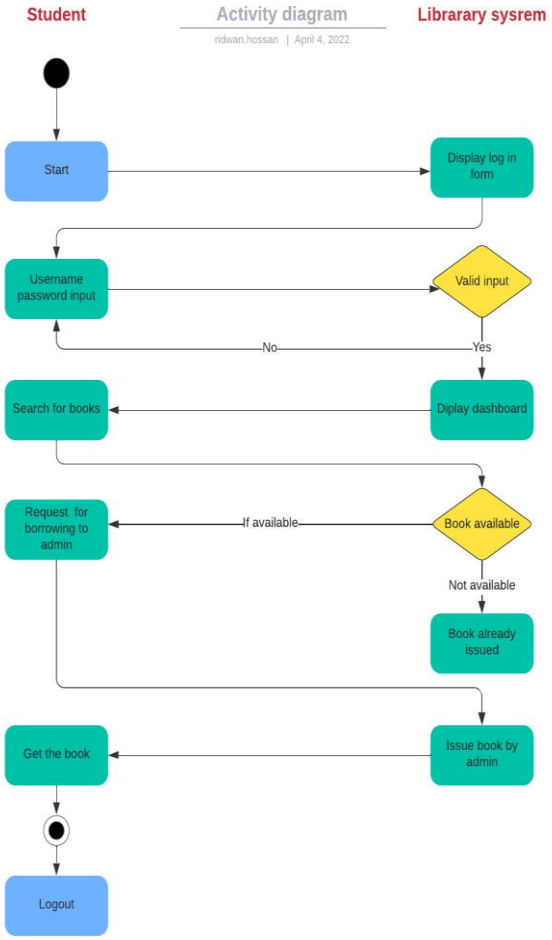


Figure 6: User activity diagram

The diagram illustrates about user activity on the library management system. According to the user activity diagram when students open the library management webpage, it shows the login form where the student have to enter personal credentials and if the credentials match with the database It displays the

user dashboard otherwise the web page return to the login form state. From the user dashboard the user can search a book and if the book is available the user can make a borrowing request to the admin, then the admin issues the book to the user. After finishing the transactions, the user has to log out from the system.

4 Graphical user interface design (GUI)

Designing a user interface is an integral part of developing any application. User interface designing is a collection of visual components that allows users to interact with computer software via a graphical user interface (GUI). A graphical user interface (GUI) is a set of components that transmit information to the user and indicate actions that the user may do. Interacting with the things causes them to alter in appearance, such as their colour, shape, and appearance. [9]

Graphical user interface is an important part for the development of a website. User experience influenced a lot on GUI. Because user don’t see the backend codes, they only see the front-end output. [9]

4.1 Home page and user login interface

This is the home page of the application. Here the student gets login form where they have to put their own credentials. In this page there is also a navigation bar which have 4 different paths, Home which redirect to home page and user log in also redirects to the same page. User signup where user have to put necessary details to sign up and Admin page redirect to admin login page.

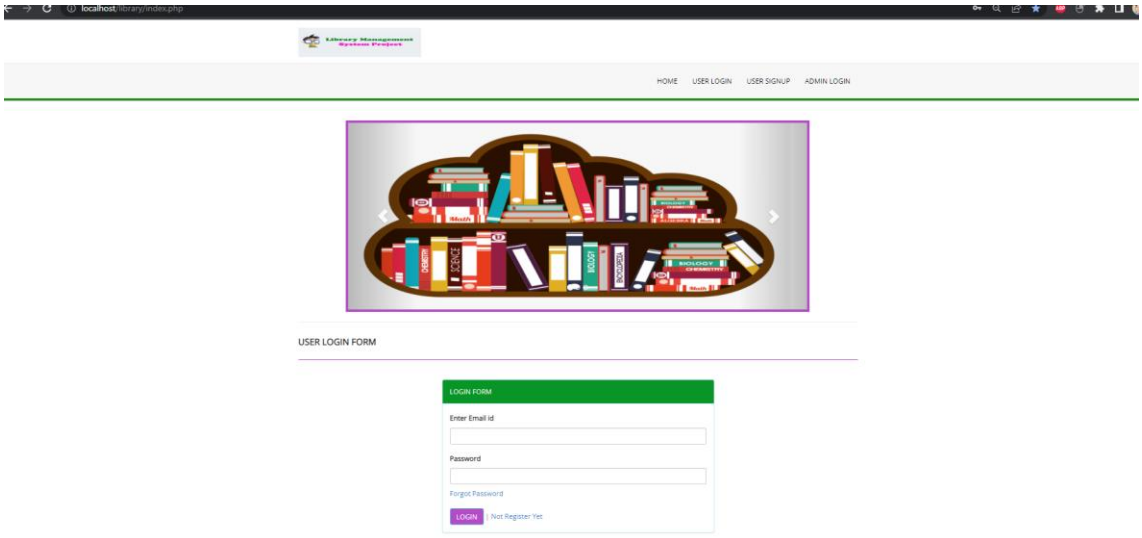


Figure 7: Home page

4.2 Admin login interface

The admin login interface displays the admin login form where the admin must put the admin credential and if it matches with admin registration details on the database it will redirect to the admin dashboard. Otherwise, the wrong username or password massage will appear.

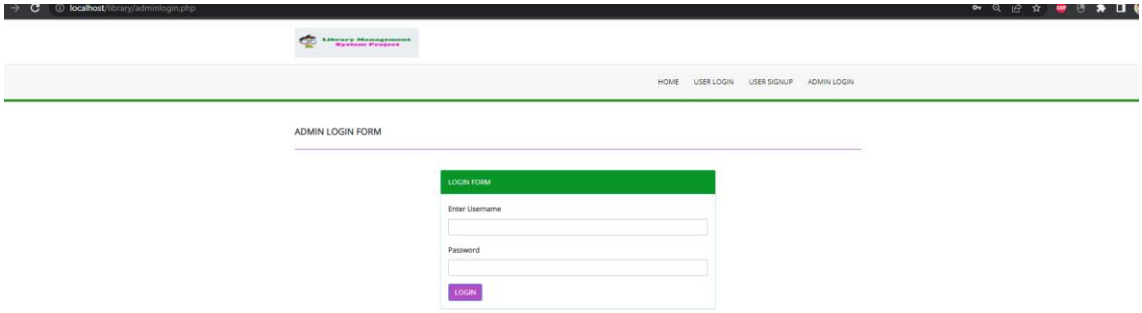


Figure 8: Admin login

4.3 User signup interface

In this interface user need to put their details (Name, email id, mobile number and password) to sign up and use the library management system. After signup the user get a unique id number.

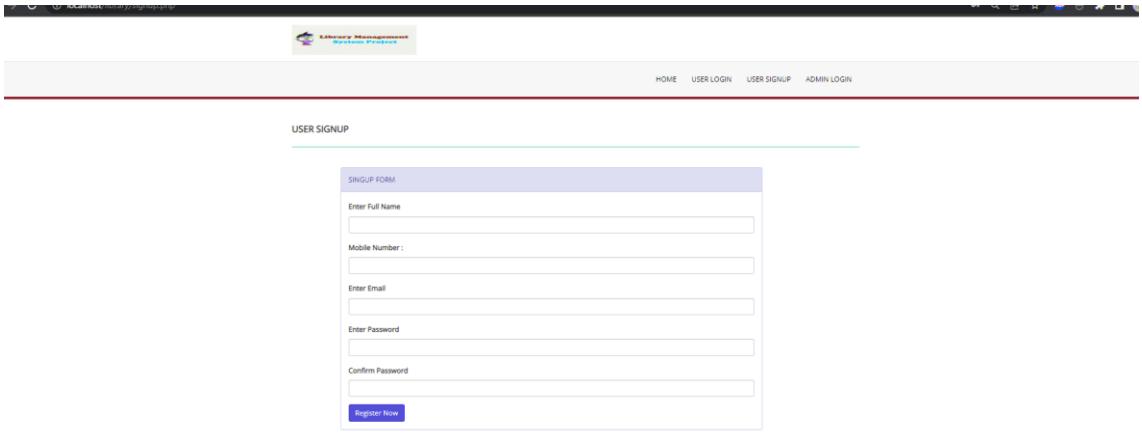


Figure 9: User signup

4.4 Admin dashboard interface

In this interface the admin can interact with various functionalities. Through this dashboard interface the admin can manage book details, issue book to the users, can track how many books have been returned, edit authors details, see the user profiles and can add fine.

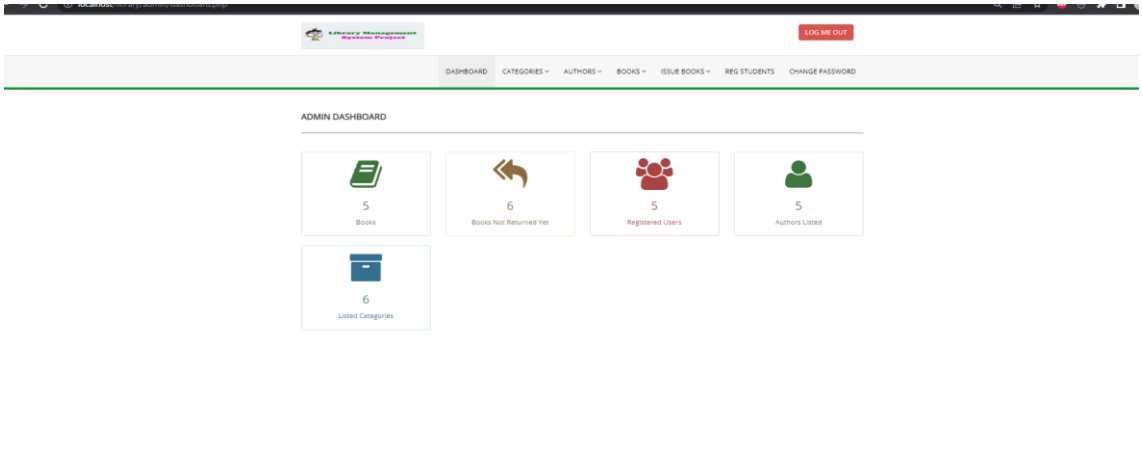


Figure 10: Admin dashboard

4.5 User dashboard interface

In this interface of the website the user can view available books in the library and also can track how many books the user borrowed form the library. From this dashboard the user can view own details and can edit own details.

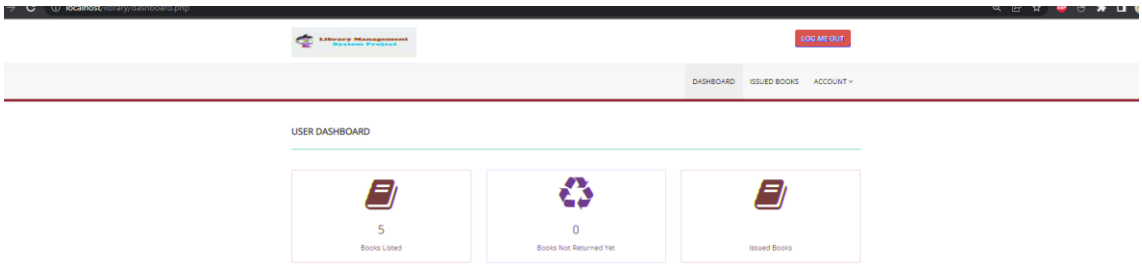


Figure 11: User dashboard

5 System Implementation

In this the section the author described about the codes and the codes functionality associated with the online LMS project.

5.1 Admin login

The bellow code checks the admin credentials (username and password) with the admin registration data on the database. If it matches the admin dashboard interface will appear. From there the admin can use different functionality of the system. And if it doesn’t match with the database data a massage will prompt invalid details.

```php
<?php
session_start();
error_reporting(0);
include('includes/config.php');
if($_SESSION['alogin']!=''){
$_SESSION['alogin']='';
}
if(isset($_POST['login']))
{
$username=$_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>";
}
}
?>
```

5.2 Signup

The below code works as a signup form were the user inputs name, mobile, number, password. After user input the data is saved into the database and the user will get a particular stuedntID with a massage states the registration is successful. And if any mandatory filled didn't properly filled a massage will appear that something is wrong, please try again.

```php
<?php
session_start();
include('includes/config.php');
error_reporting(0);
if(isset($_POST['signup']))
{

//Code for student ID
$count_my_page = ("studentid.txt");
$hits = file($count_my_page);
$hits[0] ++;
$fp = fopen($count_my_page , "w");
fputs($fp , "$hits[0]");
fclose($fp);
$StudentId= $hits[0];
$fname=$_POST['fullanme'];
$mobileno=$_POST['mobileno'];
$email=$_POST['email'];
$password=md5($_POST['password']);
$status=1;
$sql="INSERT INTO  tblstudents(StudentId,FullName,MobileNumber,EmailId,Password,Status) VALUES(:StudentId,:fname,:mobileno,:email,:password,:status)";
$query = $dbh->prepare($sql);
$query->bindParam(':StudentId',$StudentId,PDO::PARAM_STR);
$query->bindParam(':fname',$fname,PDO::PARAM_STR);
$query->bindParam(':mobileno',$mobileno,PDO::PARAM_STR);
$query->bindParam(':email',$email,PDO::PARAM_STR);
$query->bindParam(':password',$password,PDO::PARAM_STR);
$query->bindParam(':status',$status,PDO::PARAM_STR);
$query->execute();
$lastInsertId = $dbh->lastInsertId();
if($lastInsertId)
{
echo '<script>alert("Your Registration successfull and your student id is  "+"'.$StudentId.'")</script>';
}
else
{
echo "<script>alert('Something went wrong. Please try again');</script>";
}
}

?>
```

5.3 Changing password

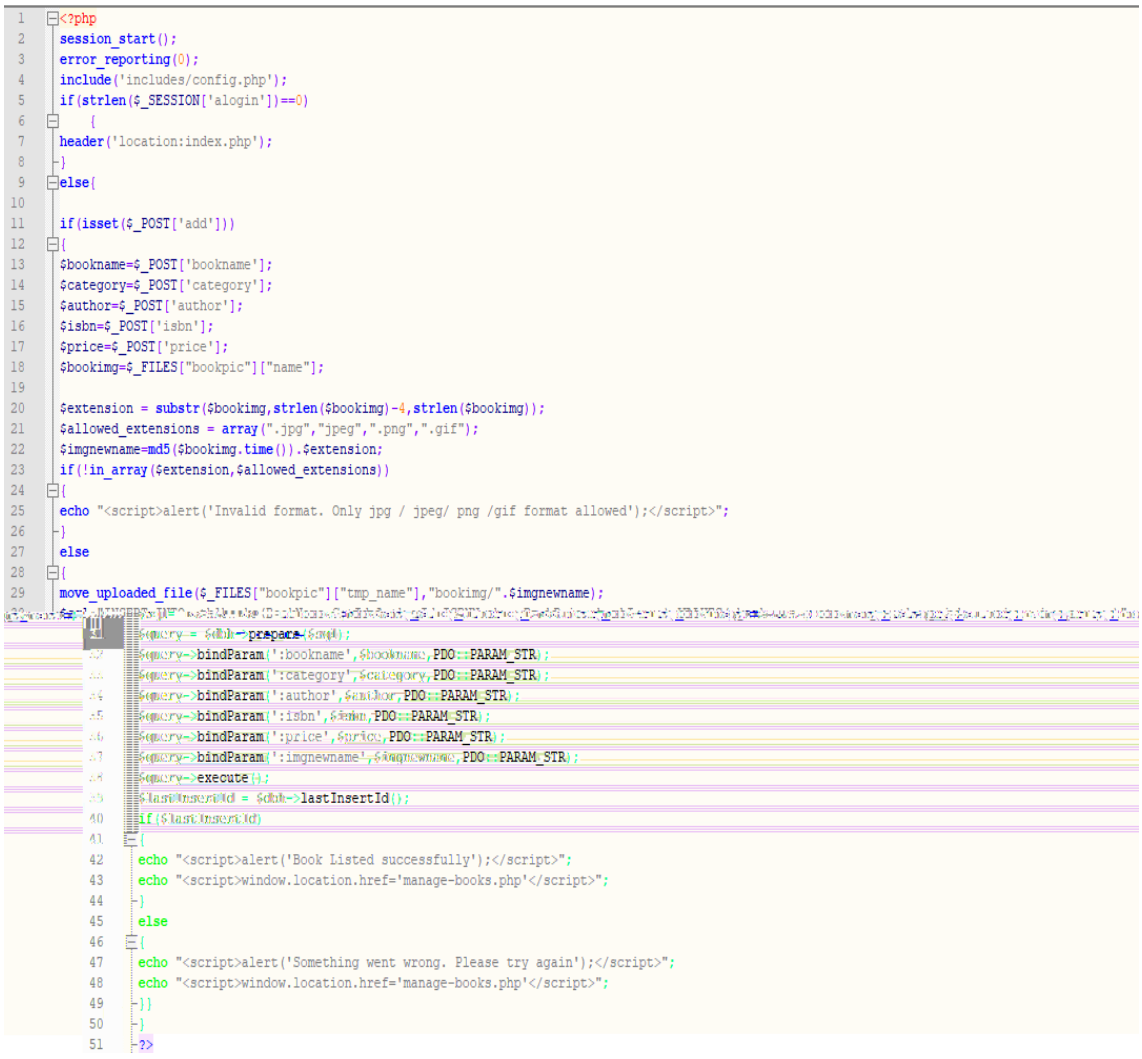
The bellow code allows the user to change the current password and set a new password. If the user doesn't input the password properly or put current password wrong, the system will prompt your current password is wrong, as the code checks the current password and it only allow changes if the current password matches with database. If the password changes successfully the system will prompt password successfully changed.

```php
<?php
session_start();
include('includes/config.php');
error_reporting(0);
if(strlen($_SESSION['login'])==0)
    {
header('location:index.php');
}
else{
if(isset($_POST['change']))
  {
$password=md5($_POST['password']);
$newpassword=md5($_POST['newpassword']);
$email=$_SESSION['login'];
  $sql ="SELECT Password FROM tblstudents WHERE EmailId=:email and Password=:password";
$query= $dbh -> prepare($sql);
$query-> bindParam(':email', $email, PDO::PARAM_STR);
$query-> bindParam(':password', $password, PDO::PARAM_STR);
$query-> execute();
$results = $query -> fetchAll(PDO::FETCH_OBJ);
if($query -> rowCount() > 0)
{
$con="update tblstudents set Password=:newpassword where EmailId=:email";
$chngpwdl = $dbh->prepare($con);
$chngpwdl-> bindParam(':email', $email, PDO::PARAM_STR);
$chngpwdl-> bindParam(':newpassword', $newpassword, PDO::PARAM_STR);
$chngpwdl->execute();
$msg="Your Password succesfully changed";
}
else {
$error="Your current password is wrong";
}
}

?>
```

5.4 Add a book

The functionality of the below code snippet allows the admin to add a book on the library database. To add a book the admin has to input book name, category, author, ISBN number, price, and picture of the book, after the input the code posts the input data in the database and after saving the book data the system prompt book listed successfully. And if mediatory input filed remain empty or filled with wrong syntax the system prompt something is wrong, please try again. The system only allows book pictures in jpg, jpeg and png format. If the uploaded picture doesn't have this kind of format extension the system will prompt invalid picture format.



```php
<?php
session_start();
error_reporting(0);
include('includes/config.php');
if(strlen($_SESSION['alogin'])==0)
    {
header('location:index.php');
}
else{

if(isset($_POST['add']))
{
$bookname=$_POST['bookname'];
$category=$_POST['category'];
$author=$_POST['author'];
$isbn=$_POST['isbn'];
$price=$_POST['price'];
$bookimg=$_FILES["bookpic"]["name"];

$extension = substr($bookimg,strlen($bookimg)-4,strlen($bookimg));
$allowed_extensions = array(".jpg","jpeg",".png",".gif");
$imgnewname=md5($bookimg.time()).$extension;
if(!in_array($extension,$allowed_extensions))
{
echo "<script>alert('Invalid format. Only jpg / jpeg/ png /gif format allowed');</script>";
}
else
{
move_uploaded_file($_FILES["bookpic"]["tmp_name"],"bookimg/".$imgnewname);
$query->bindParam(':bookname',$bookname,PDO::PARAM_STR);
$query->bindParam(':category',$category,PDO::PARAM_STR);
$query->bindParam(':author',$author,PDO::PARAM_STR);
$query->bindParam(':isbn',$isbn,PDO::PARAM_STR);
$query->bindParam(':price',$price,PDO::PARAM_STR);
$query->bindParam(':imgnewname',$imgnewname,PDO::PARAM_STR);
$query->execute();
$lastInsertId = $dbh->lastInsertId();
if($lastInsertId)
{
echo "<script>alert('Book Listed successfully');</script>";
echo "<script>window.location.href='manage-books.php'</script>";
}
else
{
echo "<script>alert('Something went wrong. Please try again');</script>";
echo "<script>window.location.href='manage-books.php'</script>";
}}
}
?>
```

5.5 Issue book to user

The output of the below code snippet is the admin issues a particular book to the user. The functionality of the code snippet is the admin have to input the student id and the ISBN number of the book which the user wants to borrow. After this input the book going to issue to that particular user's account and the system will prompt book issued successfully. If there is some mistake or the book already been issue to some other user, the system will prompt something is wrong please try again.

```php
<?php
session_start();
error_reporting(0);
include('includes/config.php');
if(strlen($_SESSION['alogin'])==0)
    {
header('location:index.php');
}
else{

if(isset($_POST['issue']))
{
$studentid=strtoupper($_POST['studentid']);
$bookid=$_POST['bookid'];
$isissued=1;
$sql="INSERT INTO  tblissuedbookdetails(StudentID,BookId) VALUES(:studentid,:bookid);
update tblbooks set isIssued=:isissued where id=:bookid;";
$query = $dbh->prepare($sql);
$query->bindParam(':studentid',$studentid,PDO::PARAM_STR);
$query->bindParam(':bookid',$bookid,PDO::PARAM_STR);
$query->bindParam(':isissued',$isissued,PDO::PARAM_STR);
$query->execute();
$lastInsertId = $dbh->lastInsertId();
if($lastInsertId)
{
$_SESSION['msg']="Book issued successfully";
header('location:manage-issued-books.php');
}
else
{
$_SESSION['error']="Something went wrong. Please try again";
header('location:manage-issued-books.php');
}

}
?>
```

5.6 Manage register users

The admin can manage register students. As example admin can see student details and if the student has overdue books the admin can make the student inactive so that the student can't borrow any books in future. All these operations are handled through this code snippet. And if the admin put wrong student id the system will prompt invalid student id, please enter valid student id.

```php
<?php
require_once("includes/config.php");
if(!empty($_POST["studentid"])) {
  $studentid= strtoupper($_POST["studentid"]);

    $sql ="SELECT FullName,Status,EmailId,MobileNumber FROM tblstudents WHERE StudentId=:studentid";
$query= $dbh -> prepare($sql);
$query-> bindParam(':studentid', $studentid, PDO::PARAM_STR);
$query-> execute();
$results = $query -> fetchAll(PDO::FETCH_OBJ);
$cnt=1;
if($query -> rowCount() > 0)
{
foreach ($results as $result) {
if($result->Status==0)
{
echo "<span style='color:red'> Student ID Blocked </span>"."<br />";
echo "<b>Student Name-</b>" .$result->FullName;
 echo "<script>$('#submit').prop('disabled',true);</script>";
} else {
?>


<?php
echo htmlentities($result->FullName)."<br />";
echo htmlentities($result->EmailId)."<br />";
echo htmlentities($result->MobileNumber);
 echo "<script>$('#submit').prop('disabled',false);</script>";
}
}
}
 else{

  echo "<span style='color:red'> Invaid Student Id. Please Enter Valid Student id .</span>";
 echo "<script>$('#submit').prop('disabled',true);</script>";
}
}



?>
```

6 System testing

System testing is a sort of testing that checks the completeness and integration of a software product before it is released to the public. In order to evaluate the system requirements from beginning to finish, a system test is performed. The majority of the time, software is a component of a larger computer system. A variety of software systems are ultimately connected to the application. When it comes to computer-based systems, system testing is a set of tests with the primary goal of putting them through their paces. [3]

In this section the author tested some basic functionality of the LMS-

6.1 Admin login testing

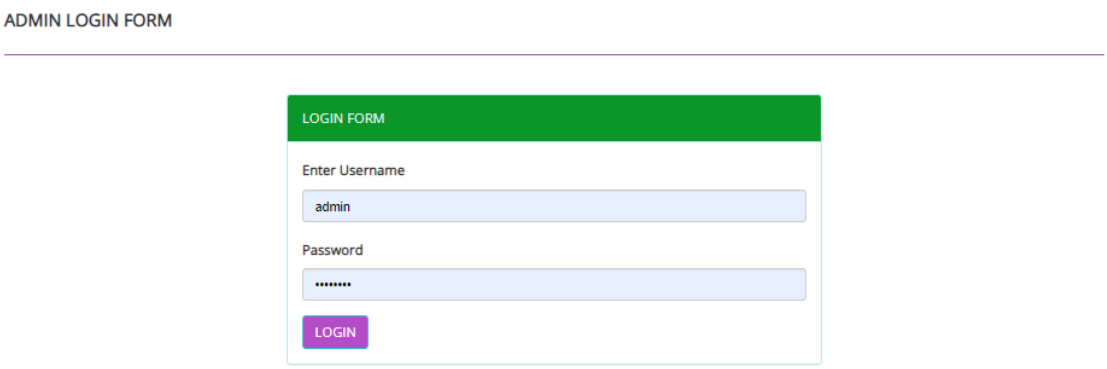


Figure 10: Admin login

Figure 12: Wrong password.

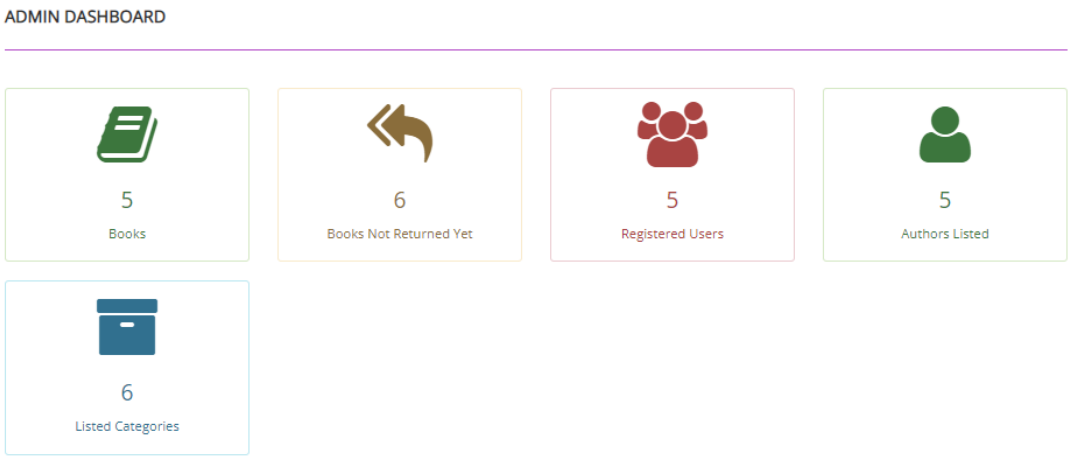


Figure 13: Login Successful.

6.2 Issue a new book functionality testing

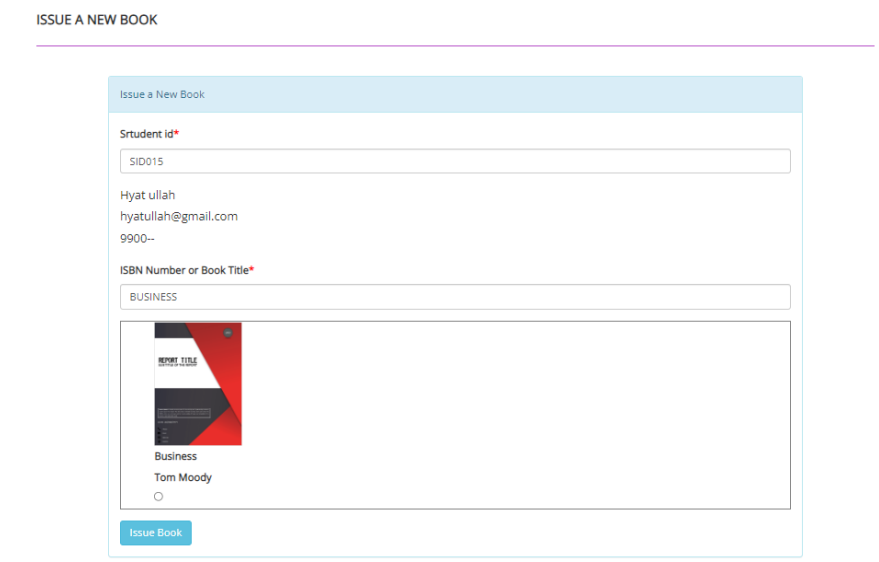


Figure 14: Issuing a new book to the user.

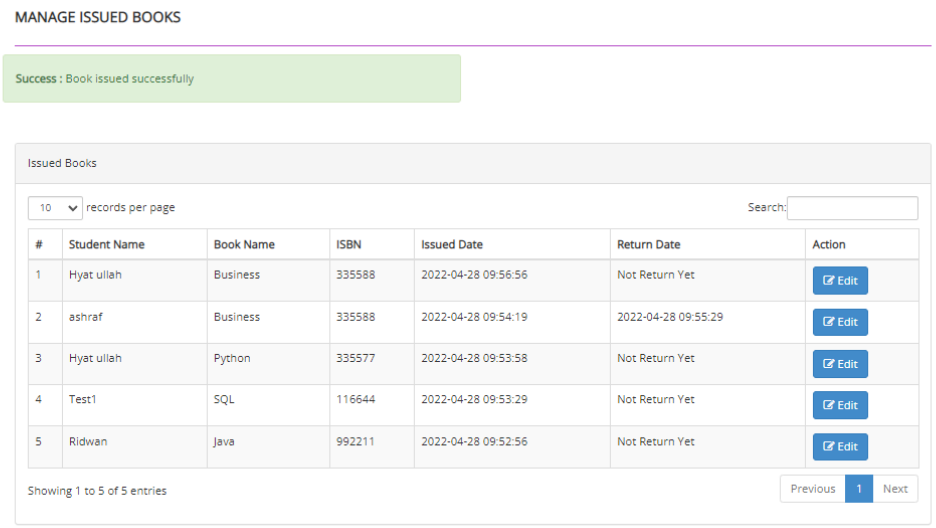


Figure 15: Successfully issued a book to the user.

6.3 User signup functionality testing

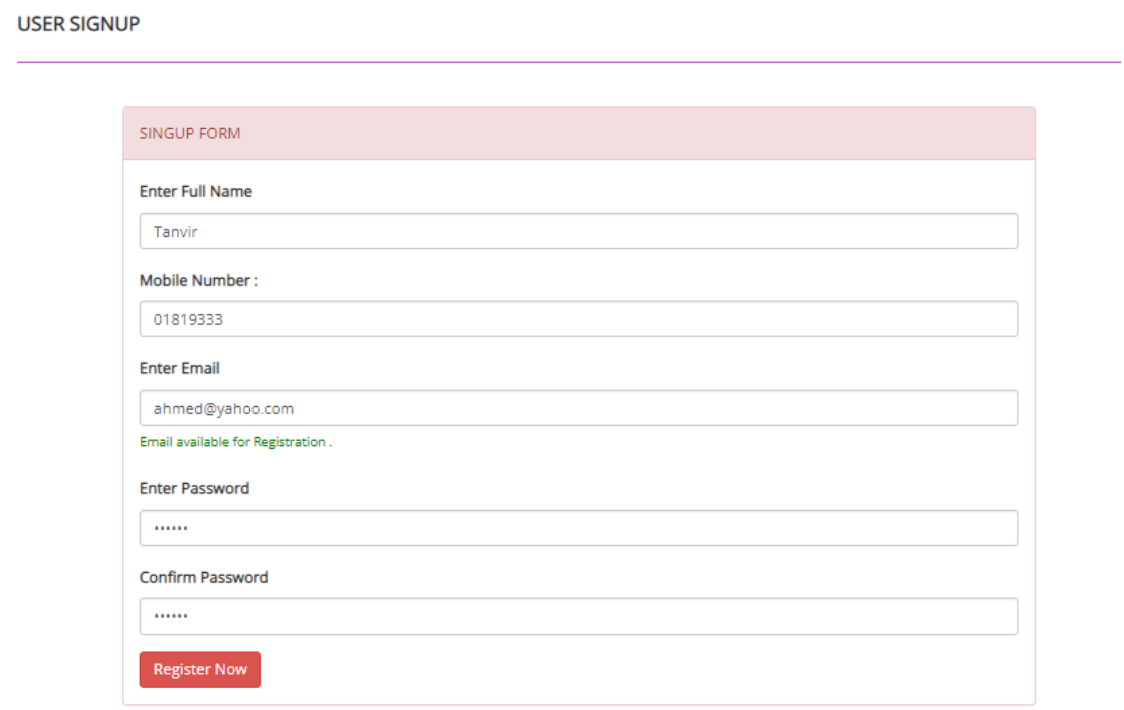


Figure 16: User signup

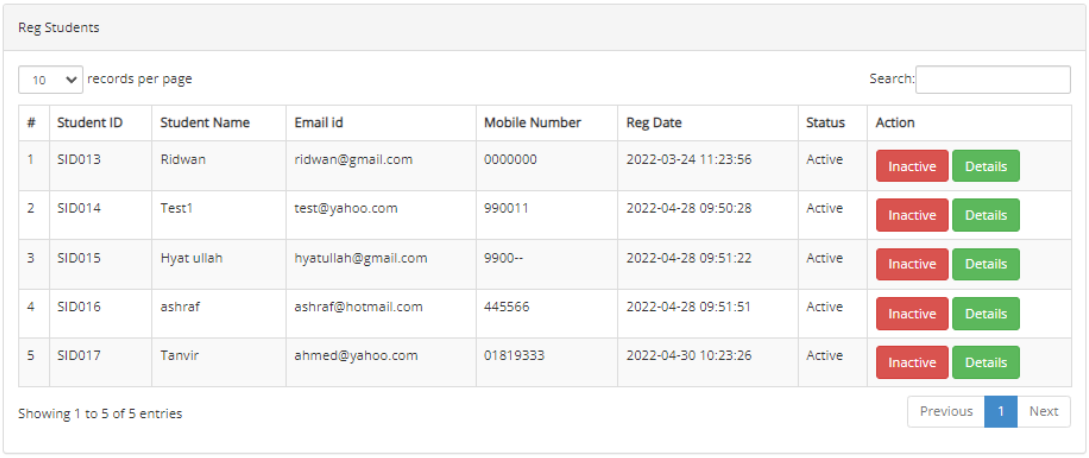


Figure 17: Student details saved in the database.

6.4 User login functionality testing

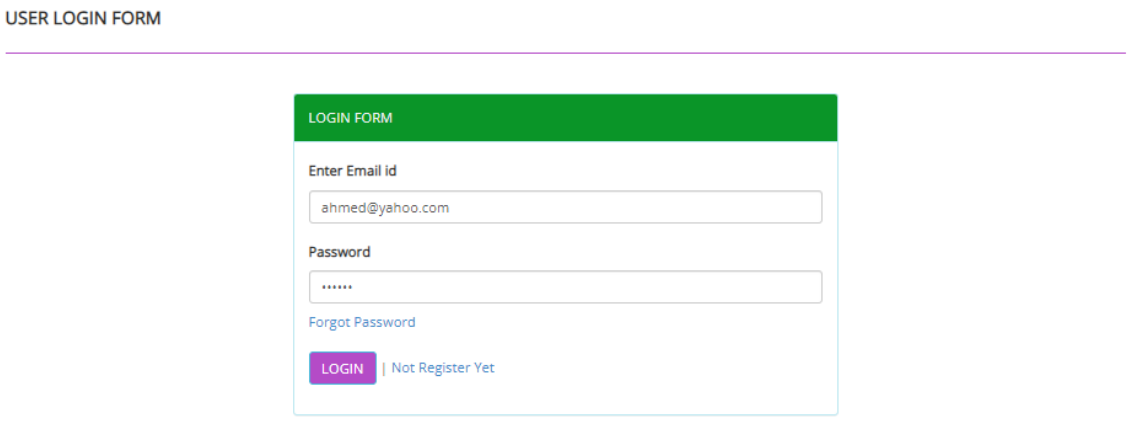


Figure 18: User login

Figure 19: Invalid password.

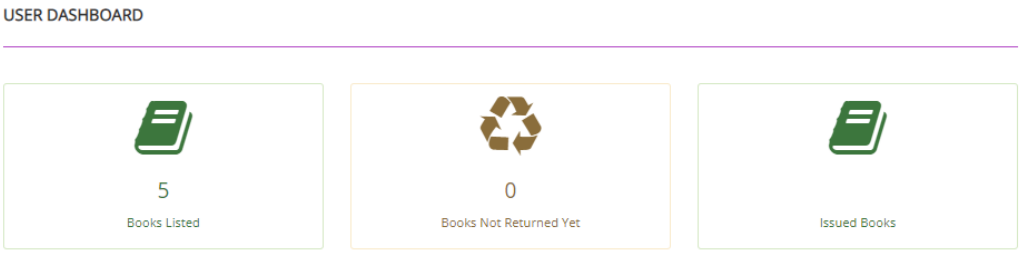


Figure 20: Login successful.

6.5 Add book functionality testing

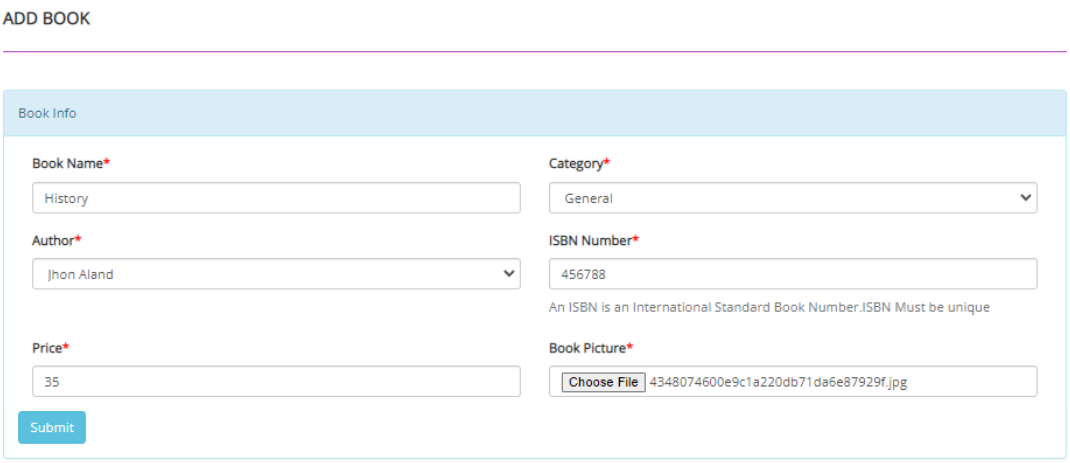


Figure 21: Adding a new book to the LMS

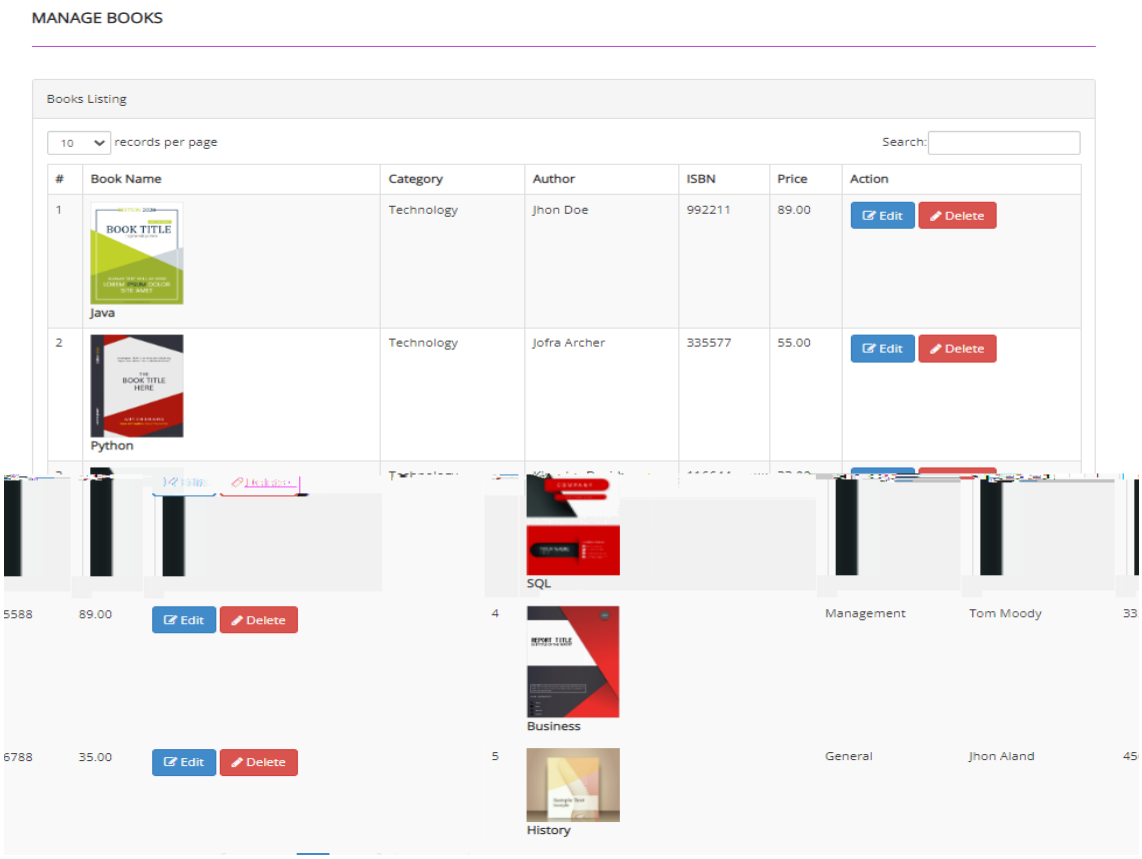


Figure 22: Book added successfully to the LMS.

7 Conclusion

The thesis project is focused on the development of a system that is both efficient and user friendly when it comes to providing library management service to the users. For achieving the goal, the author used PHP, MySQL, JavaScript bootstrap and so on. The challenging part of the project is to design and implement of the database. After overcoming the challenges, the author successfully implemented the project and tested it necessary functionalities. All the basic functionality testing is successful which the author planned to do in the LMS. The project is in protype stage and its not fully ready to actual use, as there is need to add more functionalities to ensure good user experience. As example currently the user can’t borrow a book directly by themselves. They need to put a request to the admin and the admin can only issue a book to the user. Despite its flaws, the LMS can perform the essential functions required by a library management system.

References

1 SQL Overview.
https://www.tutorialspoint.com/sql/sql-overview.htm Accessed 1 May 2022

2 Library management system.
https://www.skoolbeep.com/blog/library-management-system/ Accessed 1 May 2022

3 System testing.
https://www.guru99.com/system-testing.html Accessed 1 May 2022

4 A Review Paper on Bootstrap Framework
https://irejournals.com/formatedpaper/1701173.pdf Accessed 1 May 2022

5 Unified Modeling Language (UML)
https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-uml/#activity-diagram Accessed 1 May 2022

6 JavaScript basics
https://developer.mozilla.org/en-US/docs/Learn/Getting_started_with_the_web/JavaScript_basics Accessed 1 May 2022

7 PHP Tutorial
https://www.javatpoint.com/php-tutorial Accessed 1 May 2022

8 Oppel, A & Sheldon, R. 2009. SQL a beginner's guide. Network: McGraw-Hill.

9 Graphical user interface definition

https://www.heavy.ai/technical-glossary/graphical-user-interface Accessed 1 May 2022

10 What is MySQL?

https://dev.mysql.com/doc/refman/8.0/en/what-is-mysql.html Accessed 1 May 2022