AUTONOMOUS LIBRARY MANAGEMENT SYSTEM

Project submitted in partial fulfillment of the requirements for the award of the degree of

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING

BY

B ABHISHEK **(18C91A0509)**

C K S RUTHWIK **(18C91A0513)**

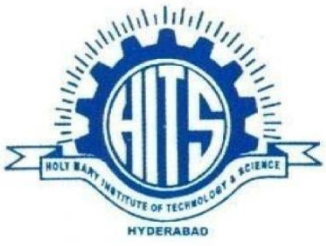
G DIKSHETHA **(18C91A0526)**

Under the Esteemed guidance of

Mrs.CH.TULASI RATNA MANI M.Tech

Assistant Professor

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING



HOLY MARY INSTITUTE OF TECHNOLOGY & SCIENCE

(COLLEGE OF ENGINEERING)

(Approved by AICTE New Delhi, Permanently Affiliated to JNTU Hyderabad, Accredited by NAAC with 'A' Grade)

Bogaram (V), Keesara (M), Medchal District -501 301.

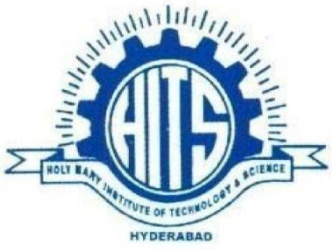
2021 - 2022

HOLY MARY INSTITUTE OF TECHNOLOGY & SCIENCE

(COLLEGE OF ENGINEERING)

(Approved by AICTE New Delhi, Permanently Affiliated to JNTU Hyderabad, Accredited by NAAC with 'A' Grade)
Bogaram (V), Keesara (M), Medchal Dist-501301.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING



CERTIFICATE

This is to certify that the mini project entitled "AUTONOMOUS LIBRARY MANAGEMENT SYSTEM" is being submitted by B.ABHISHEK (18C91A0509), C.K.S.RUTHWIK (18C91A0513), G.DIKSHETHA (18C91A0526) in Partial fulfillment of the academic requirements for the award of the degree of Bachelor of Technology in "COMPUTER SCIENCE AND ENGINEERING" from HOLY MARY INSTITUTE OF TECHNOLOGY & SCIENCE, JNTU Hyderabad during the year 2021- 2022.

INTERNAL GUIDE

Mrs.CH.TULASI RATNA MANI M.Tech
Assistant Professor
Dept. of Computer Science & Engineering

HEAD OF THE DEPARTMENT

Dr .B.NARSIMHA
Professor & HoD
Dept. of Computer Science & Engineering

EXTERNAL EXAMINER

ACKNOWLEDGEMENT

The satisfaction and euphoria that accompany the successful completion of any task would be incomplete without the mention of the people who made it possible, who's constant guidance and encouragement crowns all effort with success.

We take this opportunity to express my profound gratitude and deep regards to our Guide **Mrs.CH.TULASI RATNA MANI, Assistant Professor**, Dept. of Computer Science & Engineering, Holy Mary Institute of Technology & Science for his / her exemplary guidance, monitoring and constant encouragement throughout the project work.

Our special thanks to **Dr. B. Narsimha, Head of the Department**, Dept. of Computer Science & Engineering, Holy Mary Institute of Technology & Science who has given immense support throughout the course of the project.

We also thank Dr. **P. Bhaskara Reddy,** the **honorable Director** of my college Holy Mary Institute of Technology & Science for providing me the opportunity to carry out this work.

At the outset, we express my deep sense of gratitude to the beloved **Chairman A. Siddartha Reddy** of **Holy Mary Institute of Technology & Science**, for giving me the opportunity to complete my course of work

We are obliged to **staff members** of Holy Mary Institute of Technology & Science for the valuable information provided by them in their respective fields. We are grateful for their cooperation during the period of my assignment.

Last but not the least we thank our **Parents**, and **Friends** for their constant encouragement without which this assignment would not be possible.

B.ABHISHEK (18C91A0509)

C.K.S.RUTHWIK (18C91A0513)

G.DIKSHETHA (18C91A0526)

DECLARATION

This is to certify that the work reported in the present project titled **"AUTONOMOUS LIBRARY MANAGEMENT SYSTEM"** is a record of work done by us in the Department of Computer Science & Engineering, Holy Mary Institute of Technology and Science.

To the best of our knowledge no part of the thesis is copied from books/journals/internet and wherever the portion is taken, the same has been duly referred to in the text . The reports are based on the project work done entirely by us not copied from any other source.

B.ABHISHEK (18C91A0509)

C.K.S.RUTHWIK (18C91A0513)

G.DIKSHETHA (18C91A0526)

CONTENTS

Name of the Chapter Page No.

LIST OF FIGURES

LIST OF IMAGES

LIST OF ABBREVIATIONS

LMS	Library Management System
DBMS	Database management system
UML	Unified Modeling Language
GCP	Google Cloud Platform
AWS	Amazon Web Services
HTML	HyperText Markup Language
CSS	Cascading Style Sheets
PHP	Hypertext Preprocessor
SQL	Structured Query Language
ISO	International Organization for Standardization
EC2	Amazon's Elastic Compute Cloud
HTTP	Hypertext Transfer Protocol
HTTPS	Hypertext Transfer Protocol Secure

ABSTRACT

Library Management System is a computer software developed to manage the data of borrowing and returning books quickly and efficiently. The prevailing practises in the library are, the borrower picks up a book according to their choice and they forget to return the books by the due date. In addition to this, there is a high risk of data loss due to software corruption, human error and physical damage. In order to prevent this the project's goal is to design and develop web based software to overcome data loss using cloud services. To create borrower accounts, which notify borrowers the due date and give information about availability of a particular book, also provide an option to pre register the book.

1. INTRODUCTION

1.1. Background Motivation

In the 21^{st} century we have experienced accelerated technological advancements that enabled automation and digitalization of various services across the world, but many libraries that play key role in such advancements by providing the required knowledge are still underdeveloped and are overlooked, hence many of the libraries around the world still follow ages old methods for maintaining the libraries, though some libraries have been digitized they are still running on legacy software that are poor in design and functionality in the 21^{st} century.

AmpleTrails offers a Library Management System which is really simple to apply and fulfills all the requirements of the librarian. There are some characteristics which help librarians to make records of books, issued books, returned books, borrowed books, list of borrowers and other necessary tasks like updating the records on a daily basis. In modern organizations, where computers are automated and work according to instructions, the coordination between human beings, commodities, and computers is vital. This helps the distributors to purchase or sell the products efficiently.

1.2. Problem Statement

The current systems are mostly operated manually. The member has to first fill a membership form with all members' details. The form is then submitted, which is then entered manually into the members record. Librarian checks through the records that are manually filled to find the availability status of the book. A member can only borrow a book if the previous member has returned the book by the due date. While some libraries are digitized they have a very poor interface and lack member/user accounts and experience data corruption due to various reasons and are mostly outdated.

1.3. Project Goal

The goal of coming up with this system was to design, develop, and implement a fully library management system using cloud.

1.4. Project Objectives

- Implement Library management system using cloud.
- To develop a system that enables users to access the data remotely.
- Enable easy retrieval of books by searching the system.
- To make it easy to issue and return the books.
- Add new features like Pre registration of a book and user notification.

1.5. Existing System

The existing Library Management System (LMS) only provides the admin/librarian accounts , and are only limited to admin interfaces, they do not have user/borrower accounts, even the existing interfaces are designed poorly and are mostly outdated. The existing LMS systems are either completely dependent on the system for all it’s functionalities or are dependent on a local server, and have poor security.

1.6. Proposed System

The proposed LMS system provides both admin/librarian and user/borrower accounts with refined, sleek and latest interface and design elements. The proposed system is implemented using cloud services, and is designed in such a way that it can be implemented using most of the cloud service providers, it also provides functionalities like searching, pre registration and also provides user notifications.

1.7. Benefits of the Proposed System

"Autonomous Library Management System" is a cloud, web based Software as a Service (SaaS) designed and developed to maintain and organize any Library. This software is easy to use, its features are easy to understand and has a sleek yet simple user interface combined with new amazing features. In regular LMS softwares the borrower generally borrowed the book from the library and then mostly forgets to return it by due date, due to this the management of books in the library becomes difficult and the other borrowers who need a book cannot find the book due to its unavailability, to prevent this project adds the notification feature that notifies the user when the the return date is closing by.

2. LITERATURE REVIEW

2.1. PROJECT LITERATURE

This topic reviews information about Library Management Systems that were developed with various approaches. Throughout the world, libraries offer a conducive environment where people come to research and study. Most of these libraries are located within educational institutions, they play an important role in the operation of an institution. The main aim of the library management system is to improve the services delivered to both librarians and the members of the library.

Book issuing system was created centuries back even before the computer age. The French book wheel invention enabled scholars to circulate books by pressing a pedal that turned a table of books.The book indicator was created in 1863 by Albert Cotgreave. It held miniature versions of books that could tell if a book was available or overdue. Automation of libraries began in the 1930's, this is when the punch card systems were used to manage book acquisition and circulation.

2.2. Existing System

The existing Library Management System (LMS) only provides the admin/librarian accounts , and are only limited to admin interfaces, they do not have user/borrower accounts, even the existing interfaces are designed poorly and are mostly outdated. The existing LMS systems are either completely dependent on the system for all it’s functionalities or are dependent on a local server, and have poor security.

2.2.1. Disadvantages:

- The problem with the existing system is that all the library data (including the data of all the books and borrowers) may get lost due to physical damage of the system or due to data corruption, and low to none security protocols.
- It does not consist of borrower/user interface and accounts.
- Borrowers forget to return the books by the due date.

2.3. Proposed System

The proposed LMS system provides both admin/librarian and user/borrower accounts with refined, sleek and latest interface and design elements. The proposed system is implemented using cloud services, and is designed in such a way that it can be implemented using most of the cloud service providers, it also provides functionalities like searching, pre registration and also provides user notifications.

2.3.1. Advantages:

- Implementation of cloud services increases data security and prevents data loss .
- Also provides user profiles for users to keep track of books borrowed and approaching due dates.
- It also implements user notification and an option to pre register a book.

2.4. Applications

Autonomous Library Management Systems has various applications, it is ample and suitable for both small and large libraries. It can be used in a range of organizations, ranging from public libraries to academic libraries. This also helps the librarian keep a record of every individual book/journal of the library at the click of a button.

2.5. Summary

People have been managing libraries for centuries, although they have been responsible for providing knowledge for various technological advancements for decades yet growth and improvement of the library system has been very slow hence we implement this system that makes use of the latest cloud technology. Additionally, this system is cost-effective, so it gives institutions that cannot afford more expensive library management systems the opportunity to own one.

3. SOFTWARE REQUIREMENTS AND SPECIFICATIONS

3.1. Software requirements:

Operating System	:	Windows 7 or higher
Programming Languages	:	HTML, CSS, JavaScript, PHP, MySQL
Cloud Service Provider:	:	Azure, AWS or GCP
Browser	:	Any Browser(Chrome/Edge)

3.2. Hardware requirements:

Processor	:	Intel *i3* Core 5th Generation
RAM	:	4 GigaByte
Hard Disk	:	64 GigaByte
Peripheral Devices	:	Monitor, Keyboard, Mouse

3.3. System Feasibility:

It is during this phase that the feasibility of a project is analyzed and a business proposal is developed with a very general plan for the project and some cost estimates. During system analysis, the feasibility of the proposed system is to be evaluated. For a feasibility study, it is essential to understand the major requirements for the proposed system in order to determine whether or not it will burden the company.

Three key considerations involved in the feasibility analysis are:

- ECONOMICAL FEASIBILITY.
- TECHNICAL FEASIBILITY.
- CULTURAL FEASIBILITY.

3.3.1. Economic Feasibility

A system's economic feasibility is verified by analyzing how it will affect the organization's bottom line. A business is limited in the amount of funds it can invest in the development of a system; expenditures must be justified. Keeping this in mind we have developed this project using open source platforms, freemium softwares and AWS for hosting the website, AWS provides cloud services at a very affordable prices this system can also be hosted in a local server or in a personal computer, hence this system is economically feasible, depending on the financial status this system requires low to no cost in implementing it.

3.3.2. Technical feasibility

The Autonomous Library Management System was developed using readily used web development tools. This system is technically feasible in many ways. The system utilized a MySQL database and XAMPP Server, both open source and cross-platform. The PHP server side scripting language, Bootstrap, CSS, JQuery, and HTML5 were used for the coding.

3.3.3. Cultural Feasibility

Cultural feasibility shows us the perspective of the end users.This also includes the process of training the user to use the system efficiently, the user must not feel threatened by the system but accept the necessity of the system. The Autonomous Library Management System has a very simple and easy to use interface, and it requires no training to use this system if they have basic knowledge of using a computer or a mobile device.

4. SYSTEM DESIGN

Unified Modeling Language (UML) is used to provide a way to visualize the software design in the field of software engineering, UML is a general-purpose, developmental, modeling language, UML is managed by Object Management Group since 1997, and in 2005 UML was an approved ISO standard. UML has been evolving since the late 1990s as usage of object-oriented programming methods was improving.

UML represents the shape outline that illustrates the shape of a device by illustrating the features of the system, its characteristics, its operations, and furthermore its associations to the various groupings.

4.1. Use Case Diagram

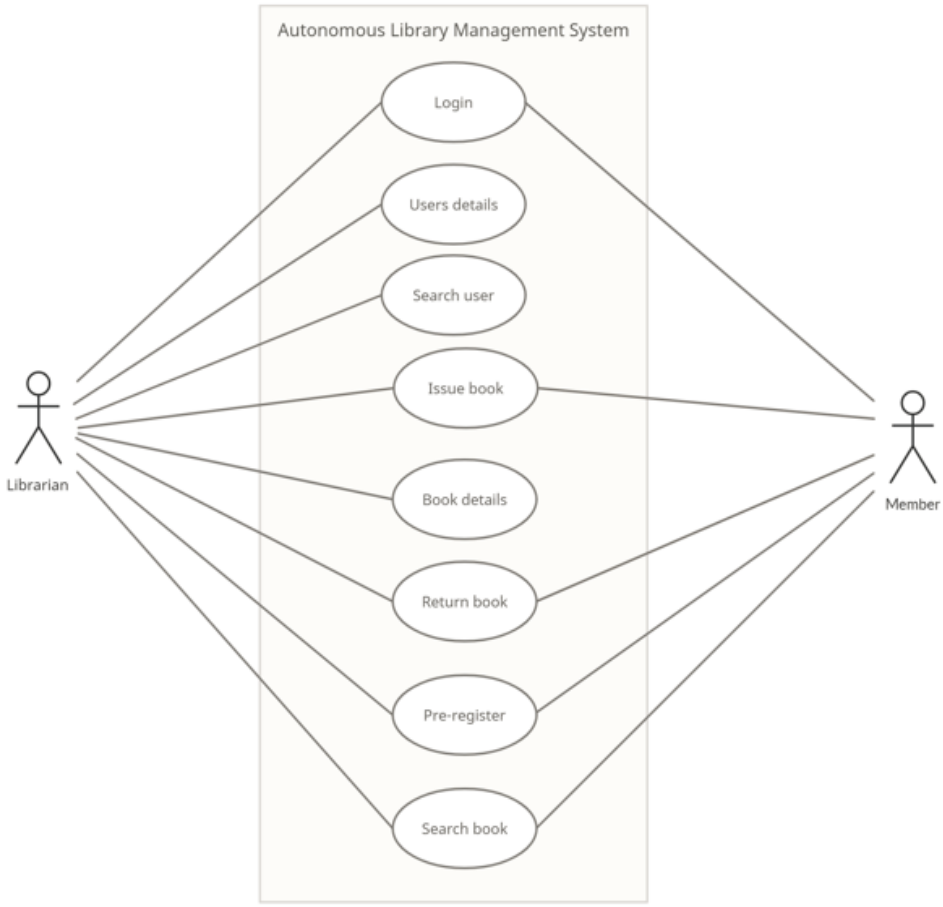


Fig 4.1: Use Case Diagram of Autonomous Library Management System

Use cases represent possible sequences of interactions between actors and systems in a particular environment, supporting a specific goal. This methodology is used in system analysis to identify, clarify, and organize system requirements. The actor can either be an external system or a human.

The Use Case diagram of the Autonomous library management system consists of two actors, namely Librarian/Admin and a Member/User and it represents various operations performed by both user and admin individually. The admin can perform operations like login, access user details, search for a user, issue books, search books and its details. The user also performs operations but are less compared to the admin. Operations performed by the user are login, return book, search for a book, and pre register a book.

4.2. Sitemap of the Autonomous Library Management System

Using a sitemap, you can describe the pages and files on your website as well as their relationships to each other. Search engines like Google, bing, and others read this file to better crawl your website for better in order to index the content of websites across the Internet so that those websites will appear in search engine results.

Autonomous Library Management System consists of mainly two sitemaps they are:

- User Portal Sitemap
- Librarian Portal Sitemap

4.2.1. User Portal

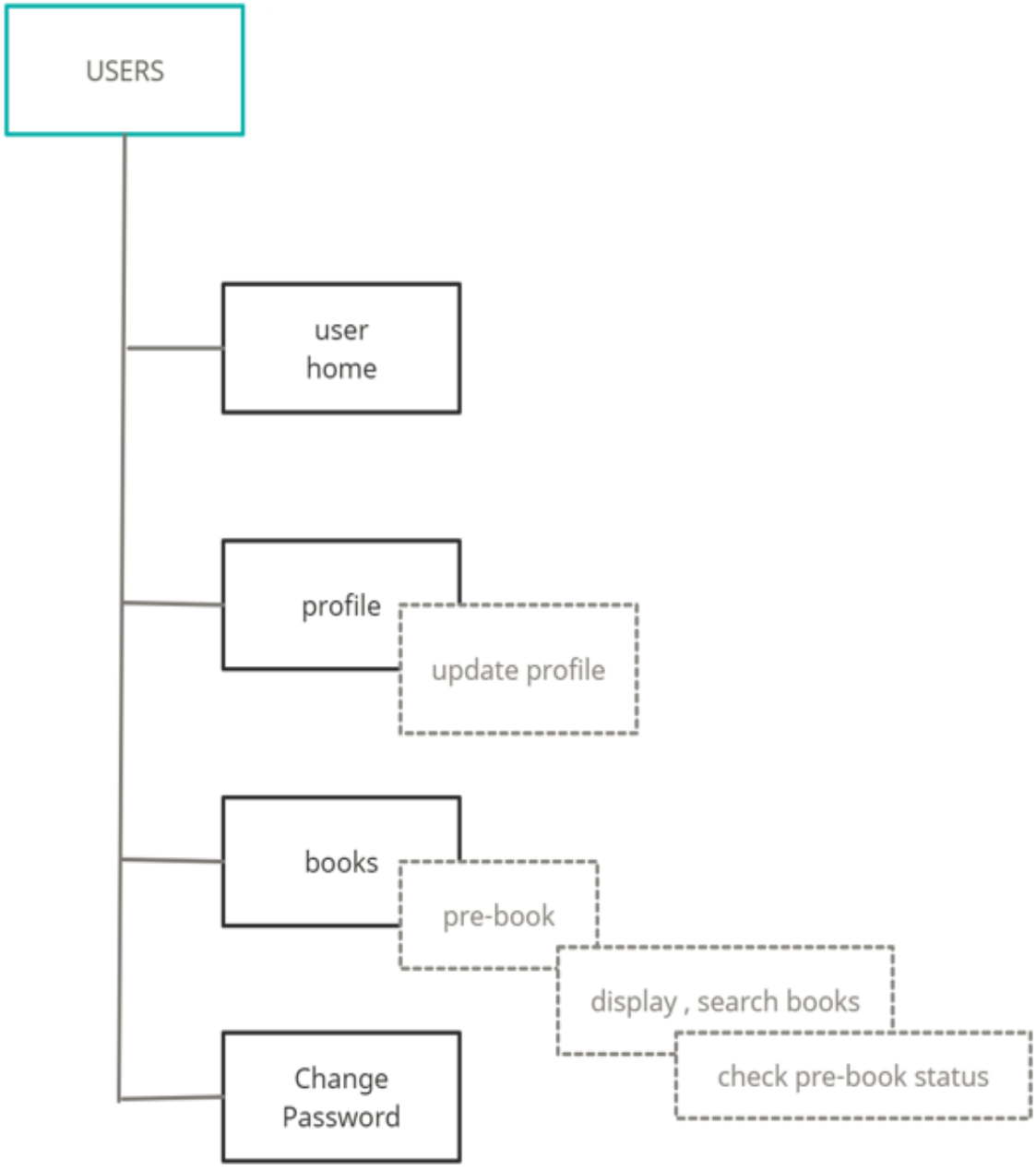


Fig 4.2: The Sitemap showing User portal

4.2.2. Librarian Portal

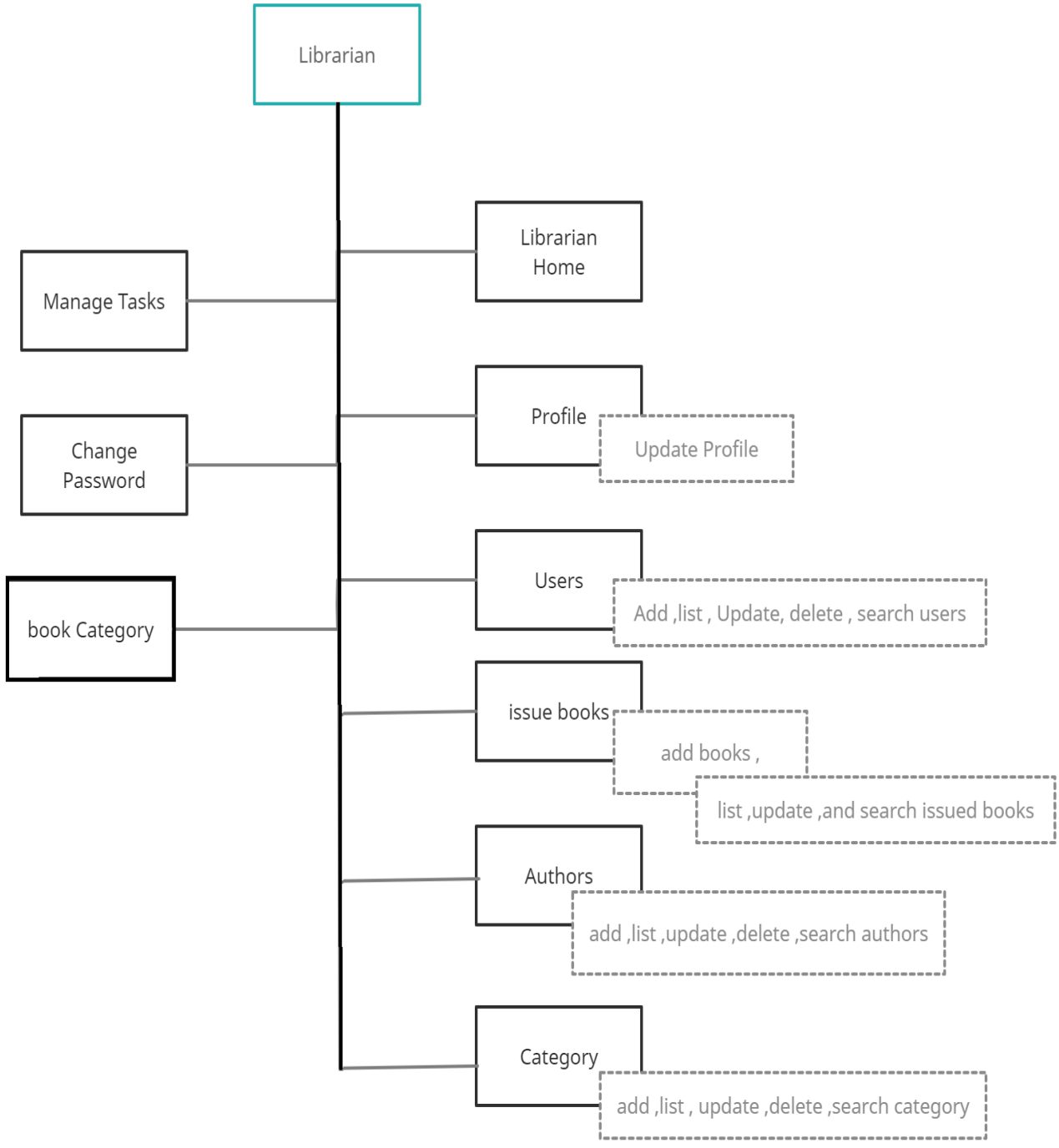


Fig 4.3: The Sitemap showing the Librarian portal

4.3. Sequence Diagrams

Sequence diagrams are interaction diagrams that display how objects interact, in which order, by describing how they're connected together, following are various sequence diagrams that show various sequences of messages passed between various objects .

4.3.1. Login Sequence Diagram

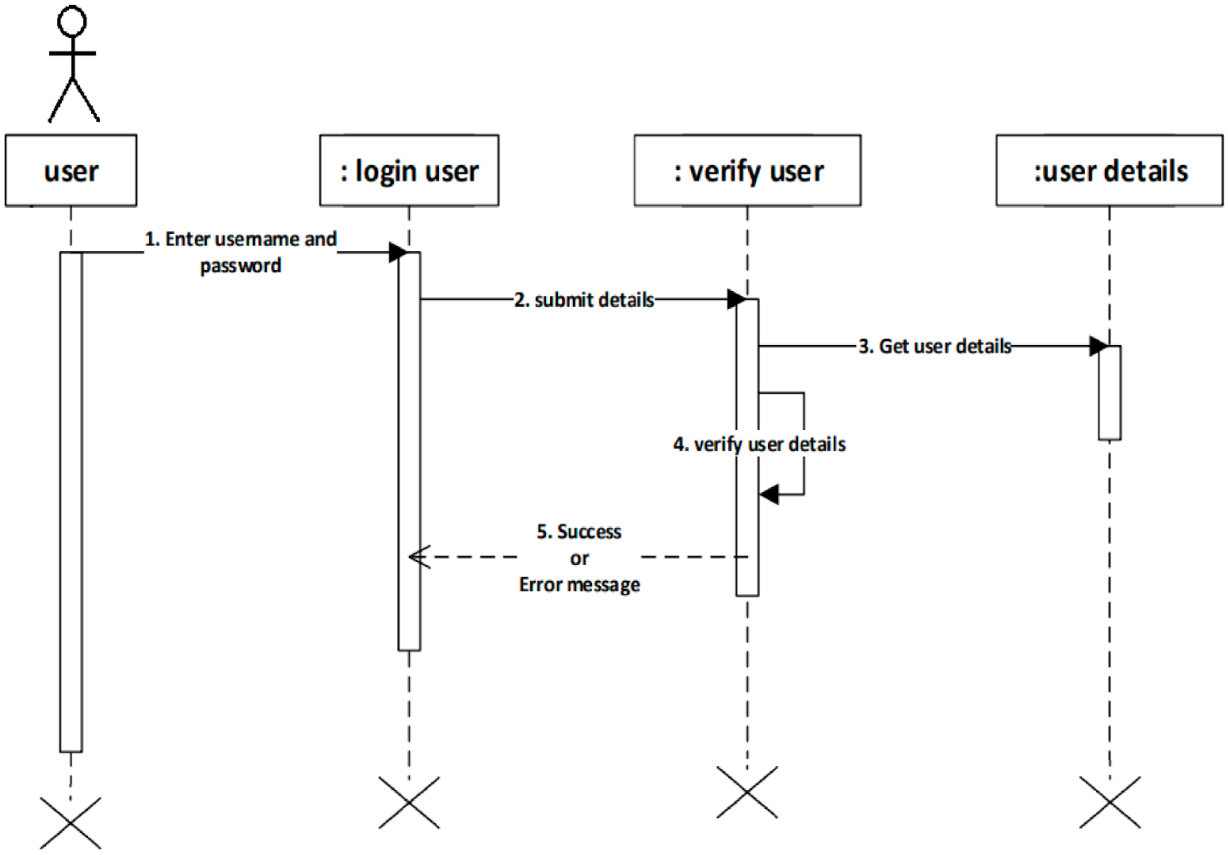


Fig 4.4: Login Sequence Diagram

Process:

1. Enter username and password.
2. Submit details.
3. Get user details from the user's data base
4. Verify user details.
5. Issue a success or error message.

4.3.2. Issue book Sequence Diagram

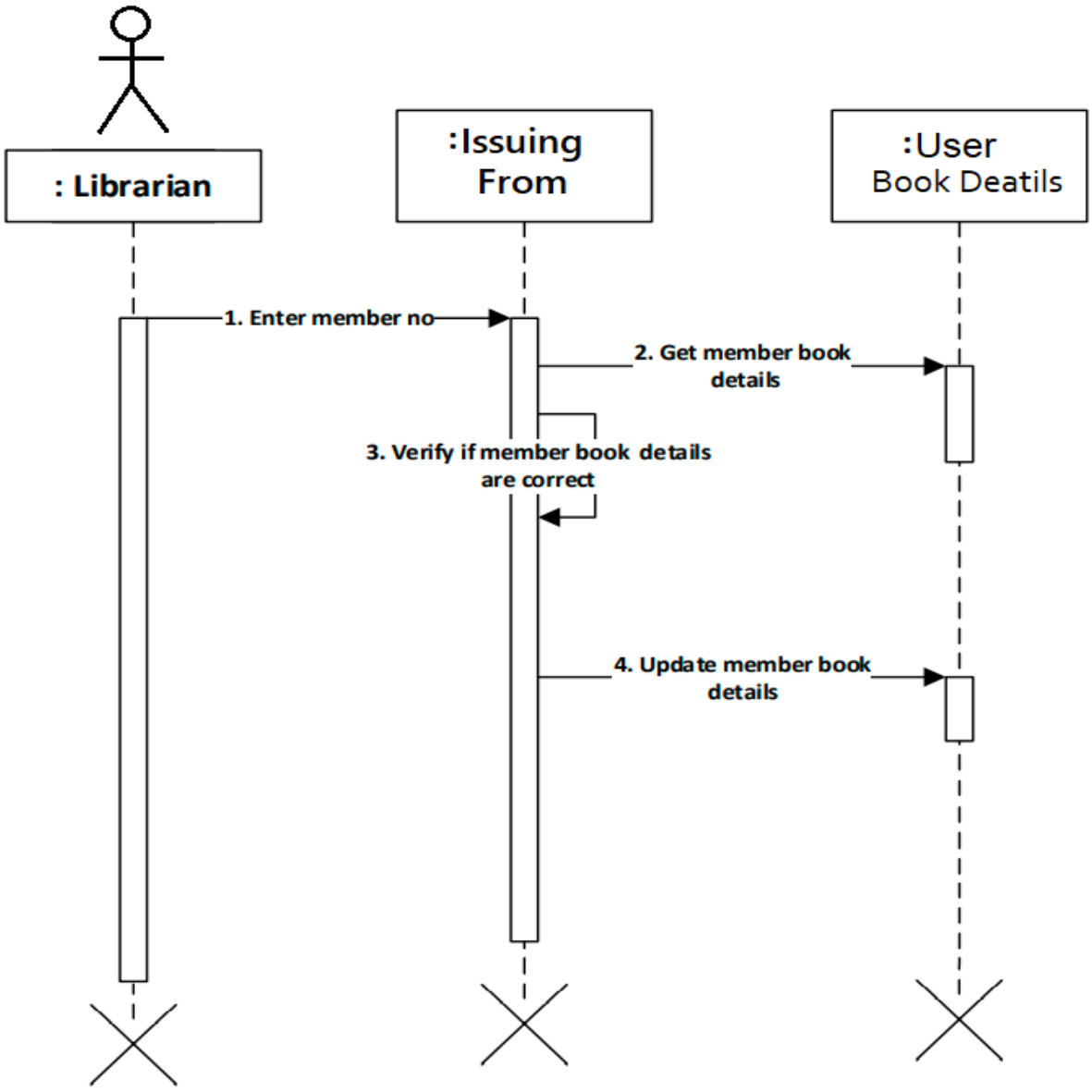


Fig 4.5: Issue book Sequence Diagram

Process:

1. Enter member number.
2. Get member book details.
3. Verify if member book details are correct.

4.3.3. Search Book Sequence Diagram

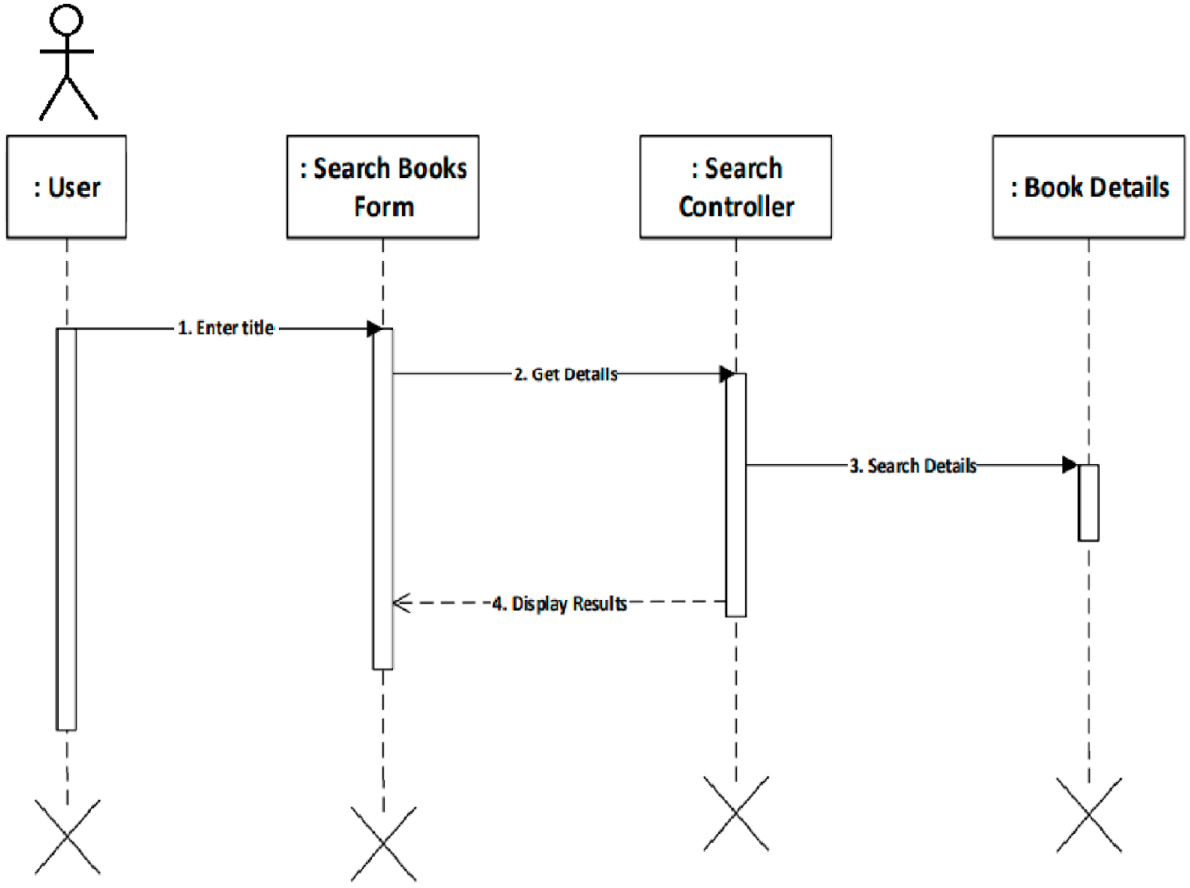


Fig 4.6: Search Book Sequence Diagram

Process:

1. Enter book title.
2. Submit details.
3. Search details.
4. Display book.

4.3.4. Search User Sequence Diagram

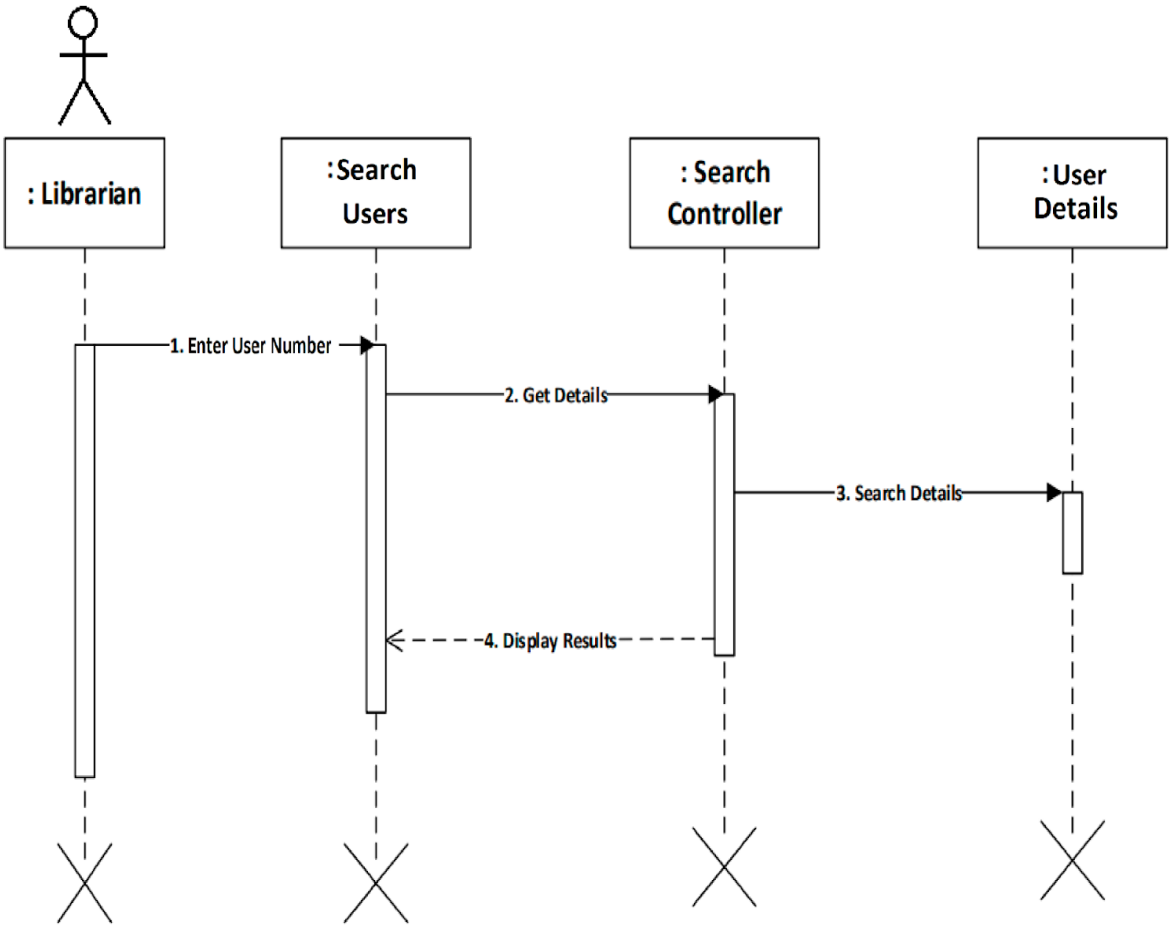


Fig 4.7: Search User Sequence Diagram

Process:

1. Enter user number.
2. Submit details.
3. Search details.
4. Display results

4.3.5. Return book Sequence Diagram

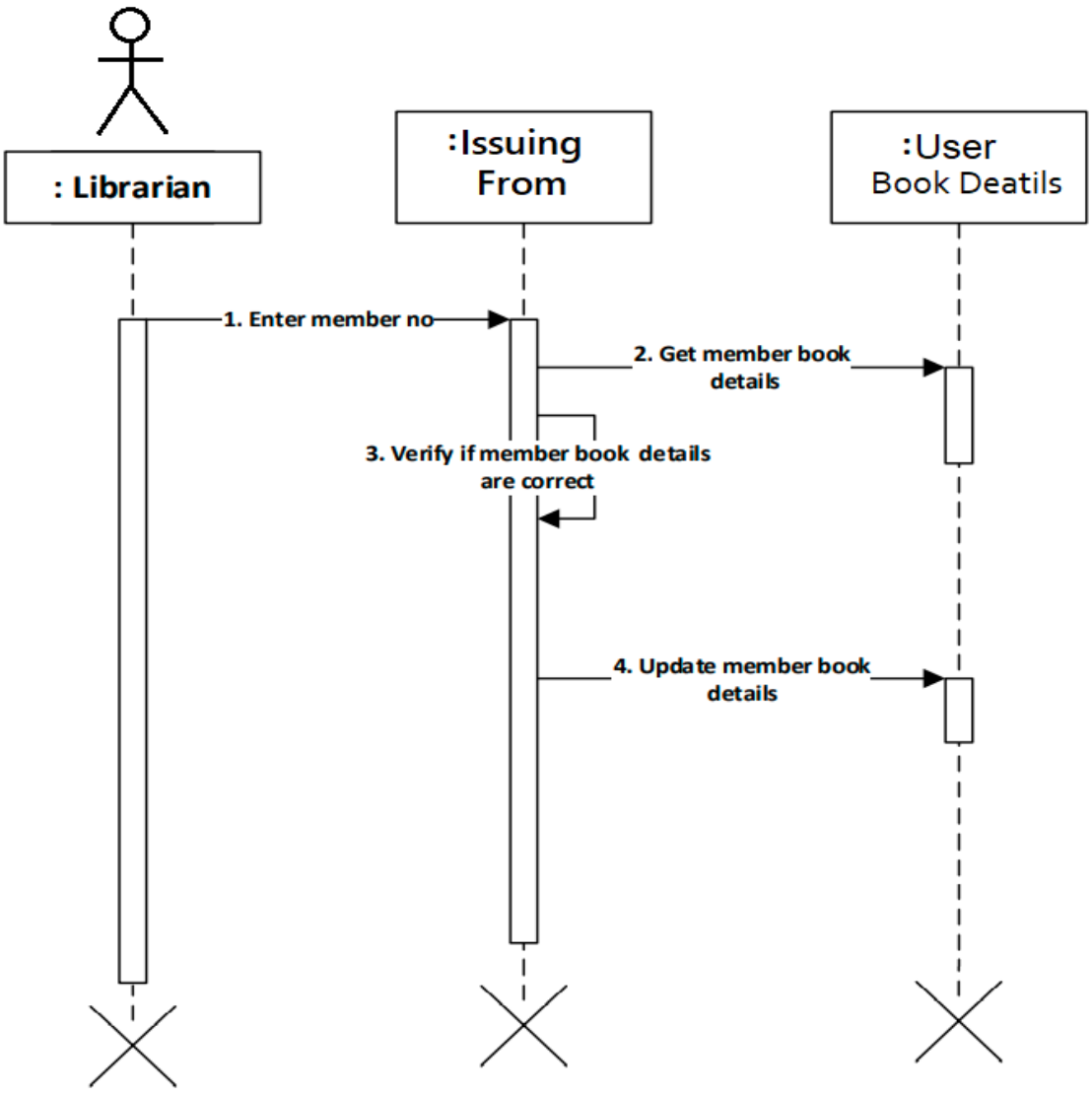


Fig 4.8: Return book Sequence Diagram

Process:

1. Enter member number.
2. Get member book details.
3. Verify if member book details are correct.

4.3.6. Pre Registration of a Book Sequence Diagram

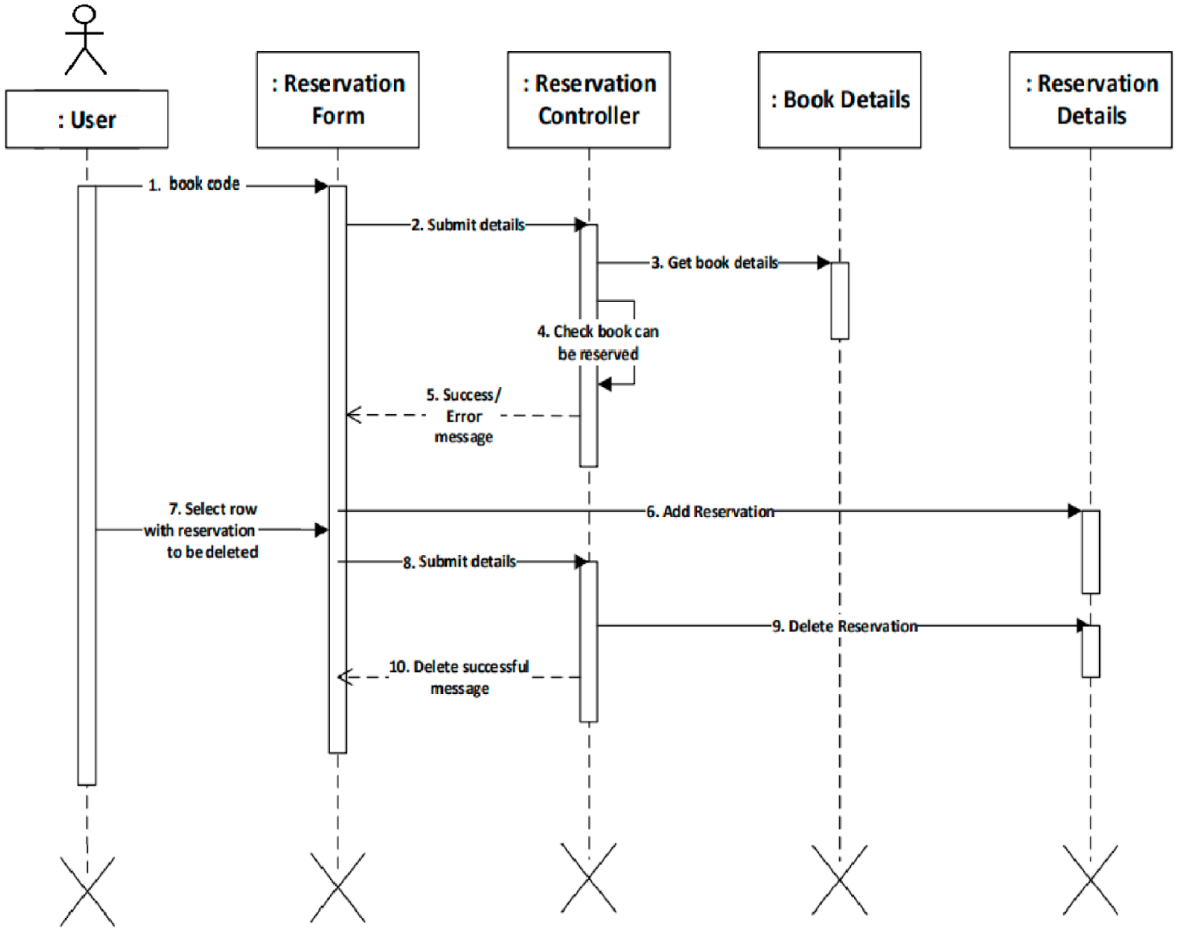


Fig 4.9:Pre Registration of a Book Sequence Diagram

Process:

1. Enter book code.
2. Submit details.
3. Get book details.
4. Check books can be reserved.
5. Success or error message.
6. Add reservation.
7. Select row with reservation to be deleted.
8. Submit details.
9. Delete reservation.
10. Success message.

4.4. Class Diagram

Class Diagrams are only UML diagrams that can be directly mapped with object oriented languages. Class diagrams are used in modeling of object oriented systems, they are also called blueprints of a system, as they describe various relations between objects and services that they provide.

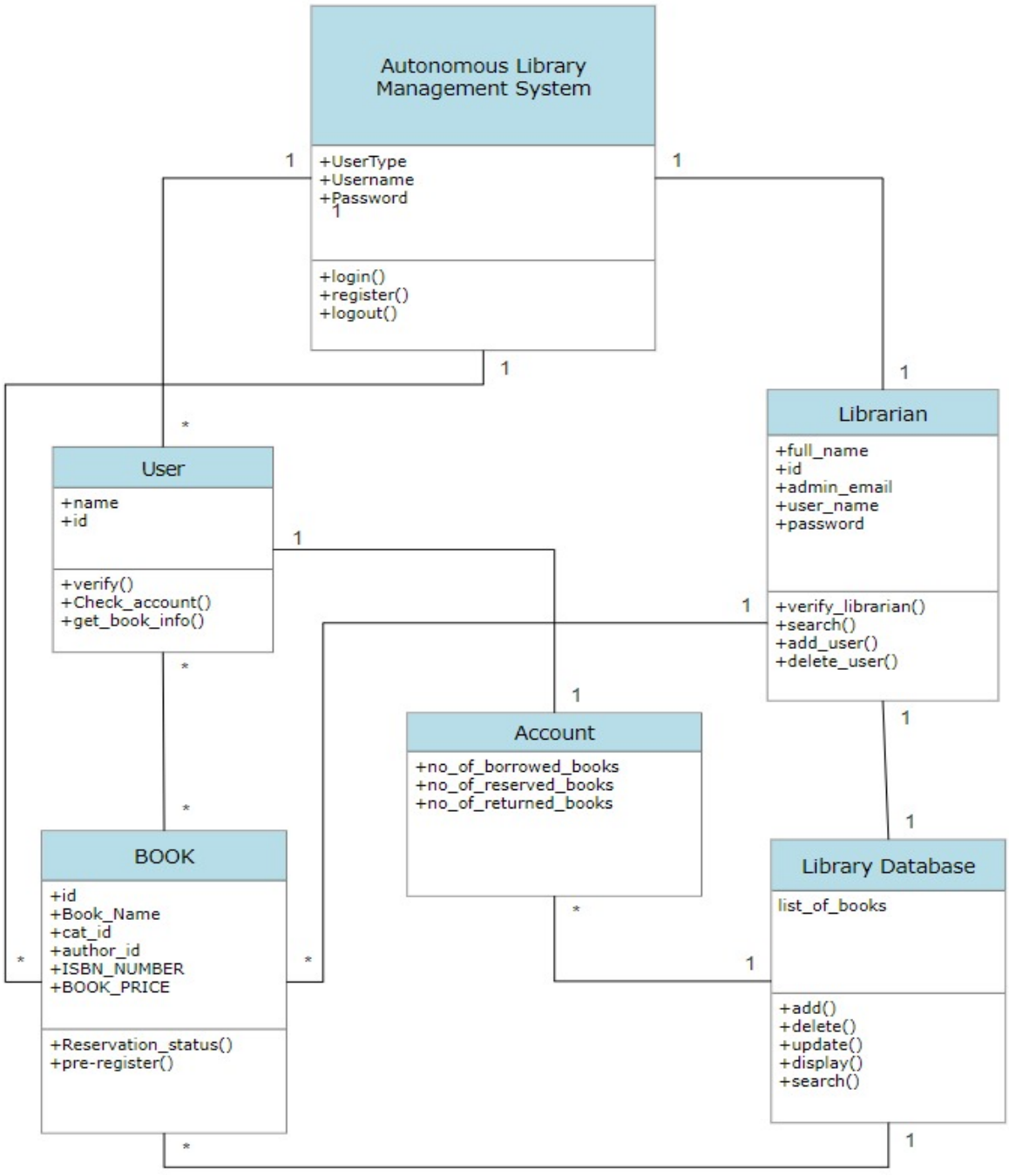


Fig 4.10:Class Diagram

4.5 Activity Diagrams

Activity diagram portrays the control flow showing the various decision paths that exist while an activity is being executed between start point to finish point. Activity diagram describes and depicts the behavior of a system.

4.5.1 Login Activity Diagram

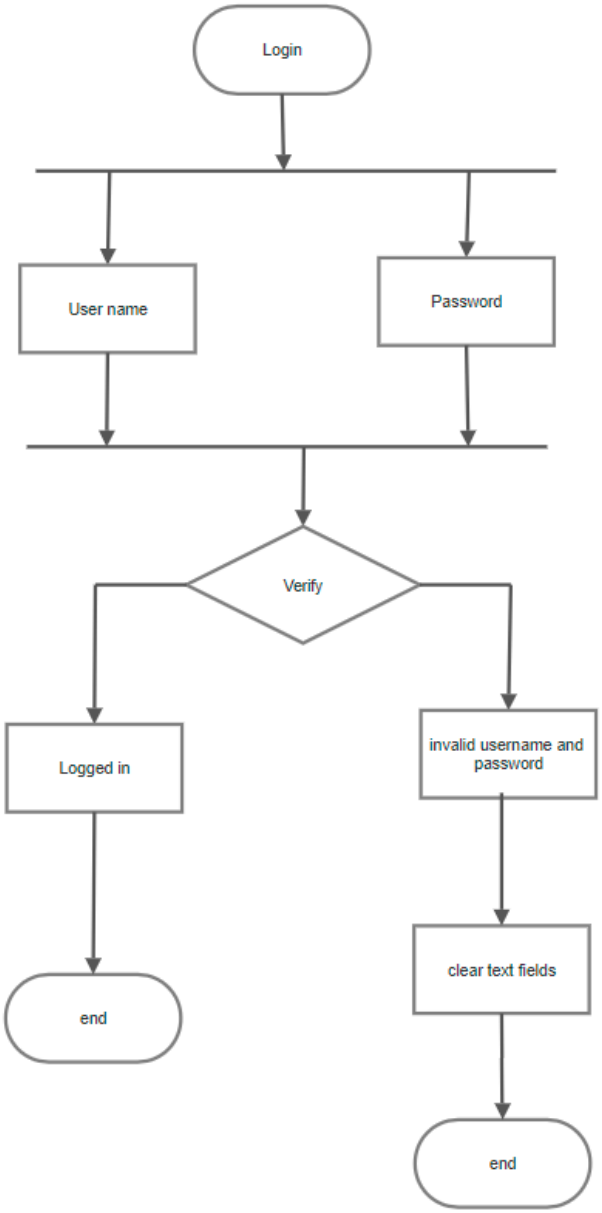


Fig 4.11: Login Activity Diagram

4.5.2. Maintain Books Activity Diagram

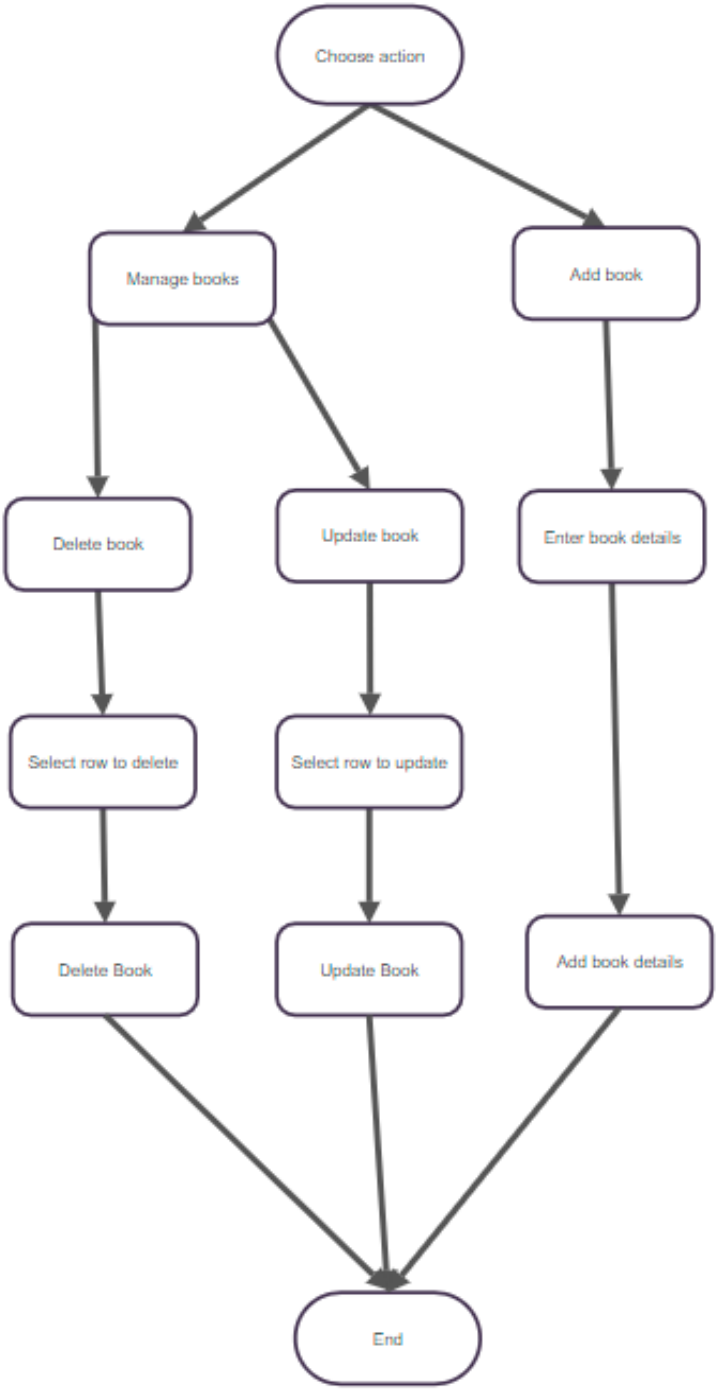


Fig 4.12: Maintain Books Activity Diagram

4.5.3. Issue Books Activity Diagram

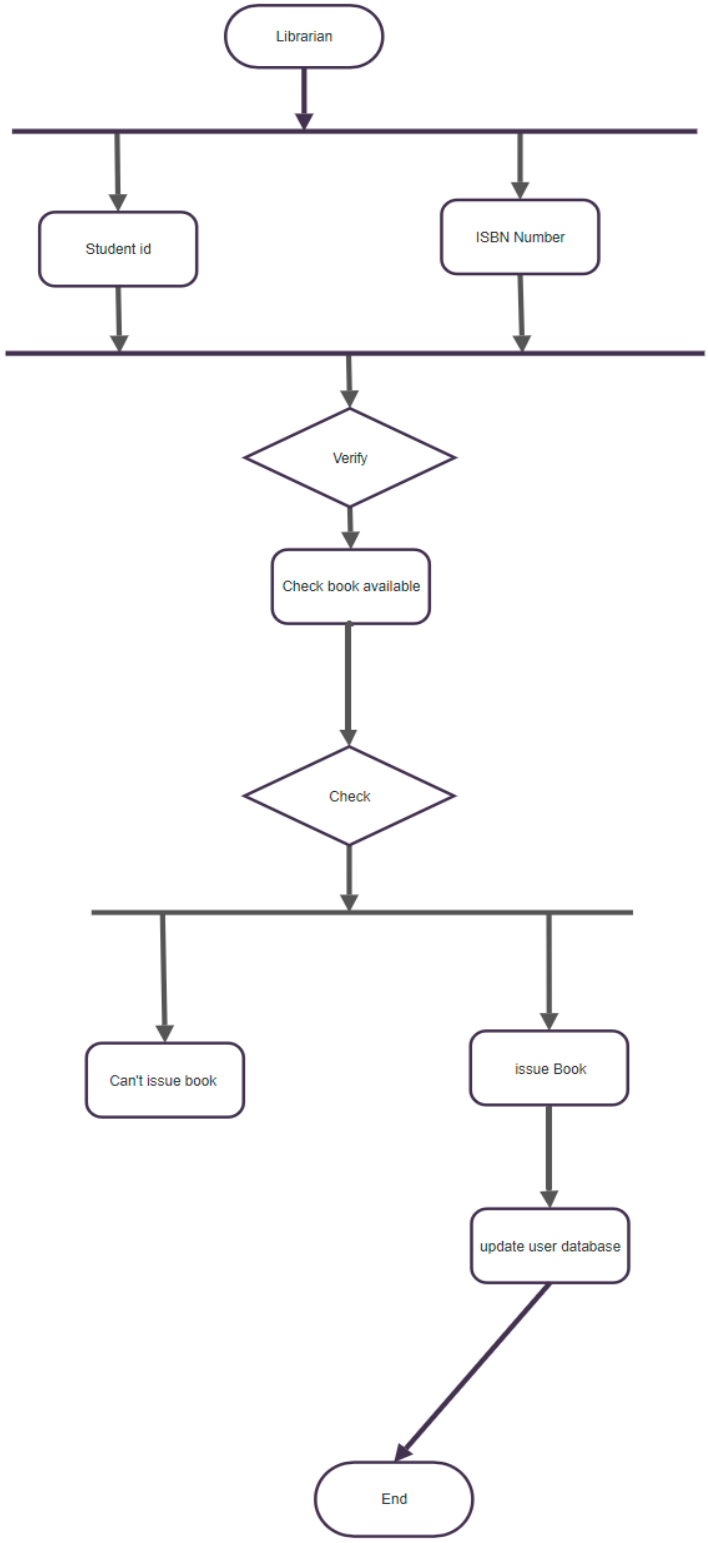


Fig 4.13: Issue Books Activity Diagram

4.5.3. Search and Pre register Books Activity Diagram

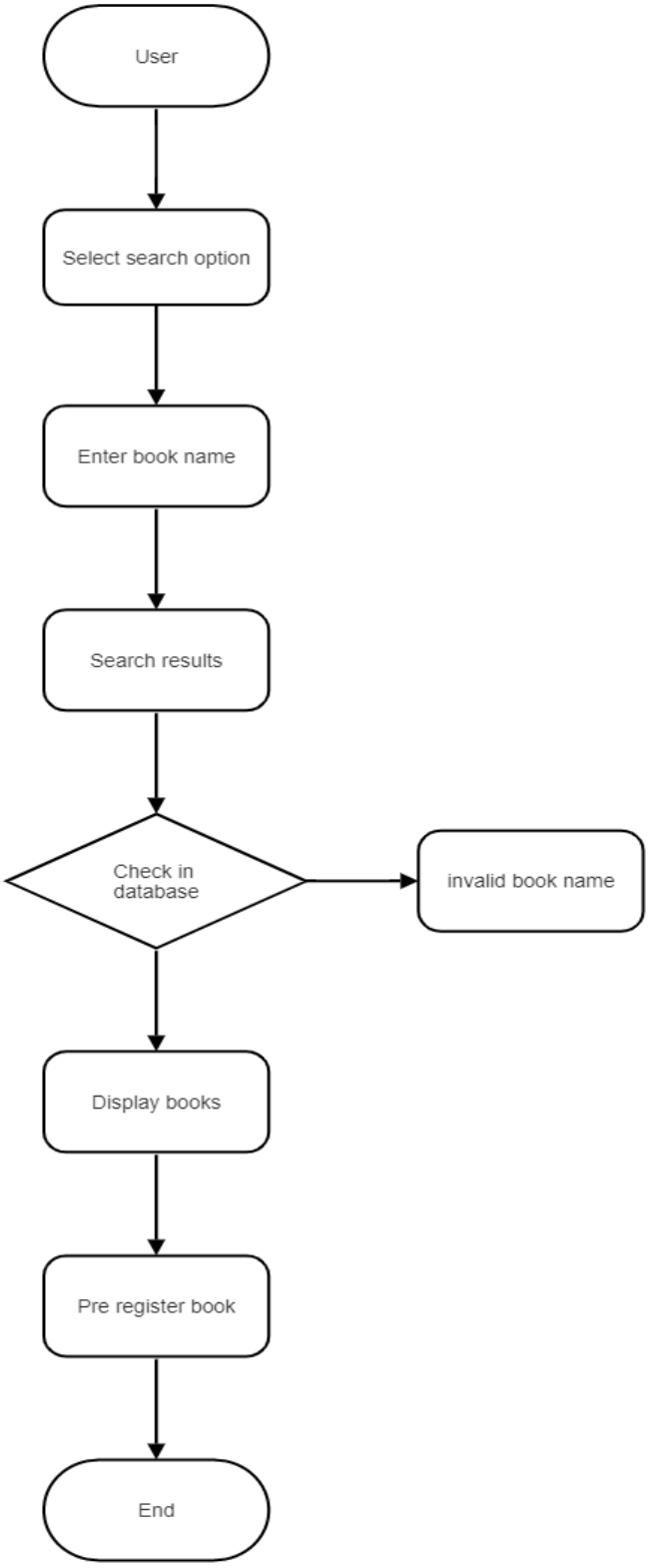


Fig 4.14: Search and Pre register Books Activity Diagram

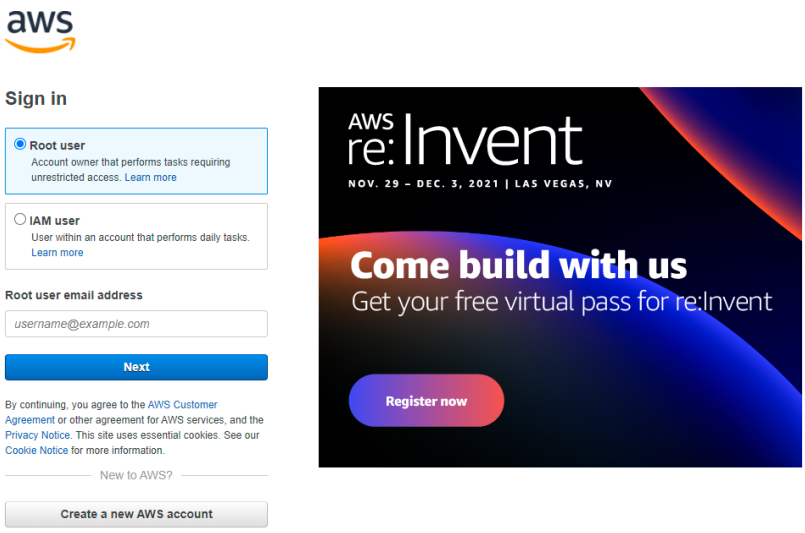
5. IMPLEMENTATION AND RESULTS

5.1. Environmental Setup

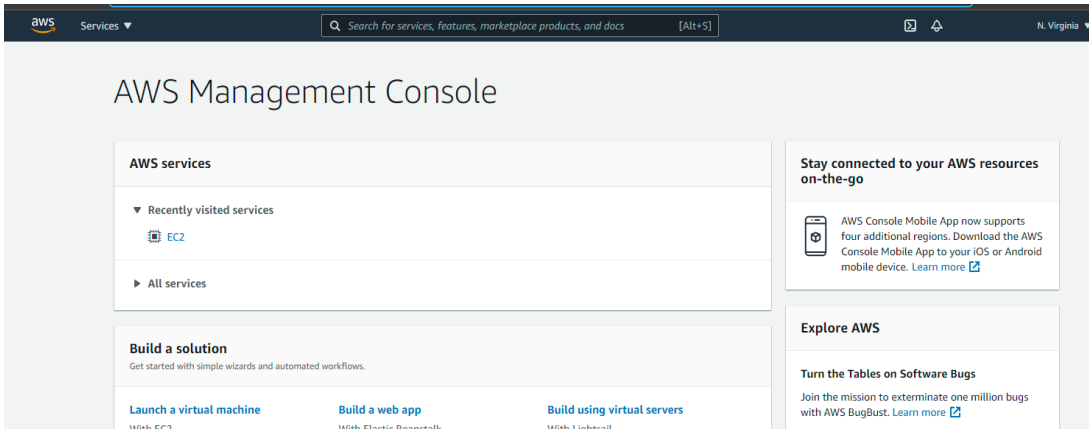
In environmental setup we will study how we will use AWS and setup a windows cloud server and how we host the website using AWS cloud.

Step 1 : Open the AWS website using the link and login into the root account.

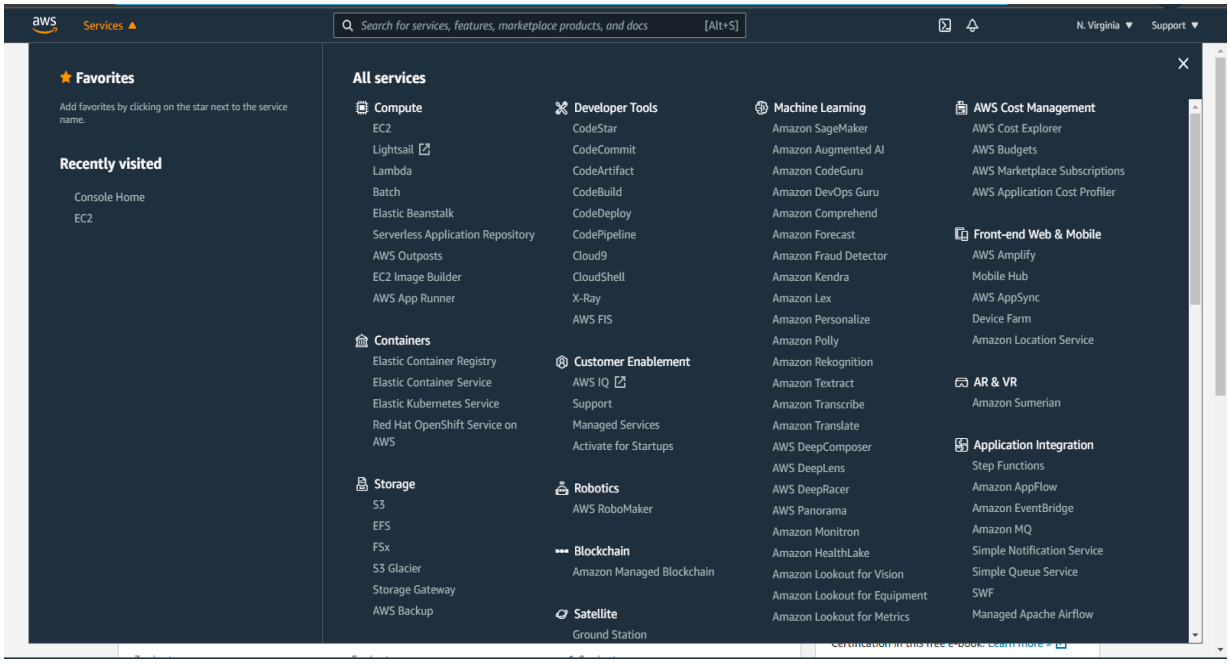
Link: *https://console.aws.amazon.com/*



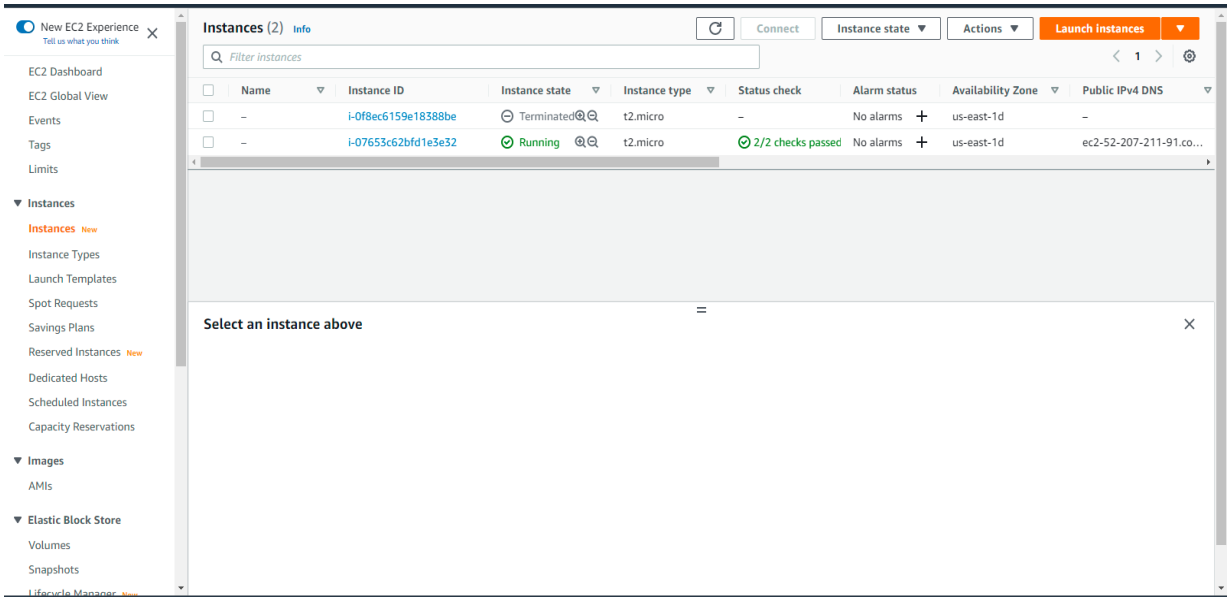
Step 2 : Click on “Services.”



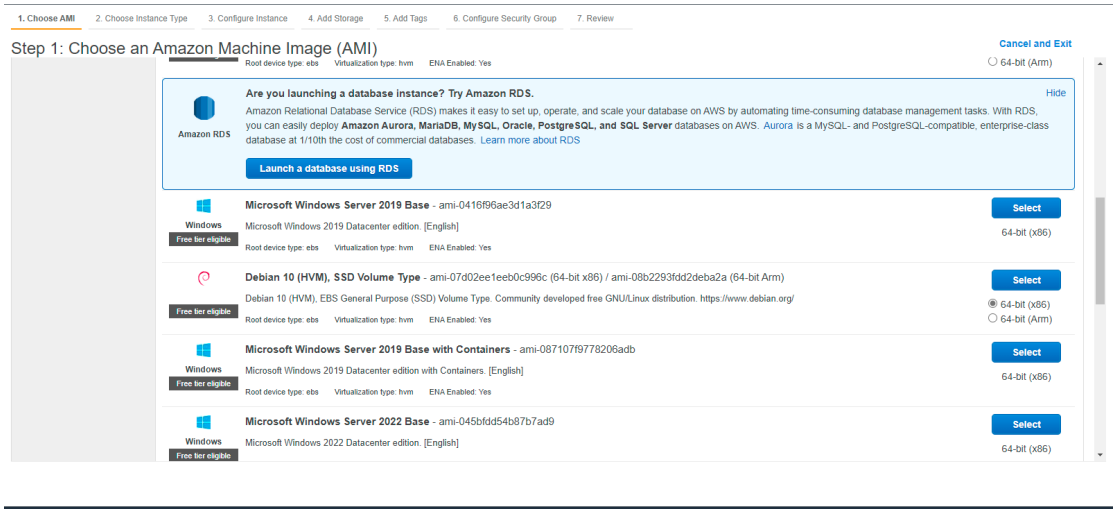
Step 3 : Under “Compute” select “EC2.”



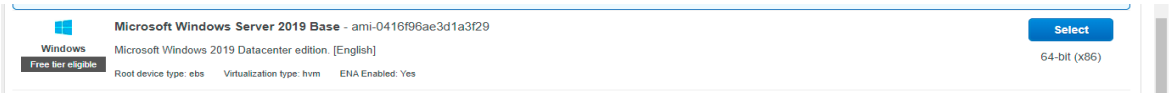
Step 4 : Click on “Launch Instance.” to create a new instance.



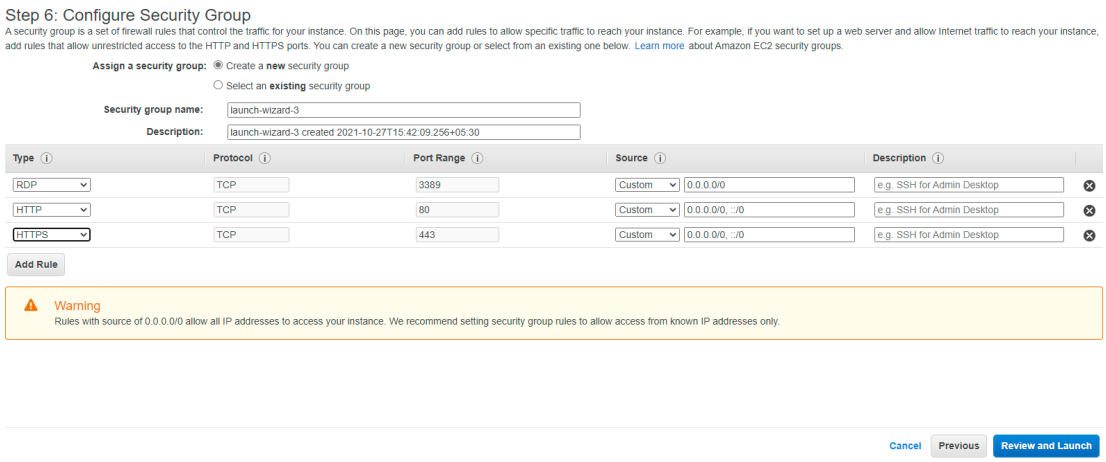
Step 5 : Choose a server to host the website(windows preferred).



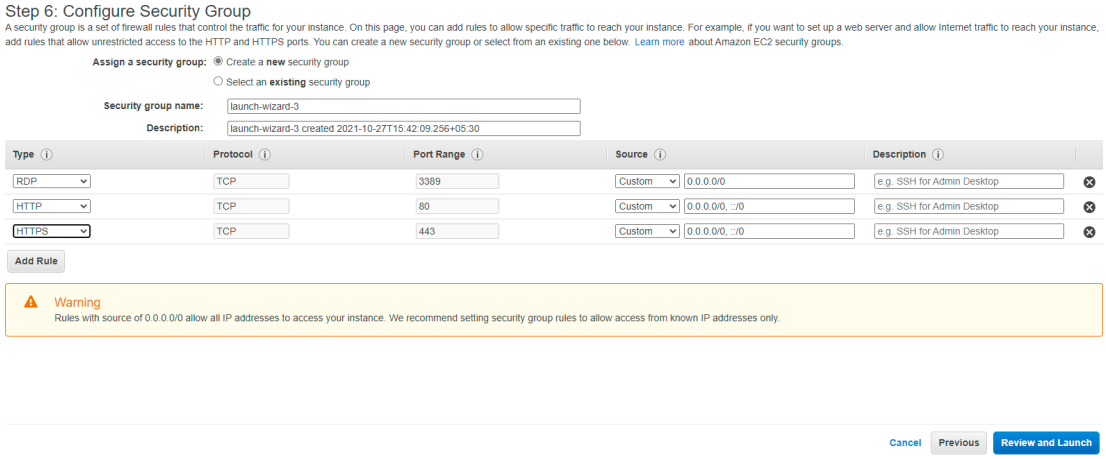
Step 6 : Click on “Select.”



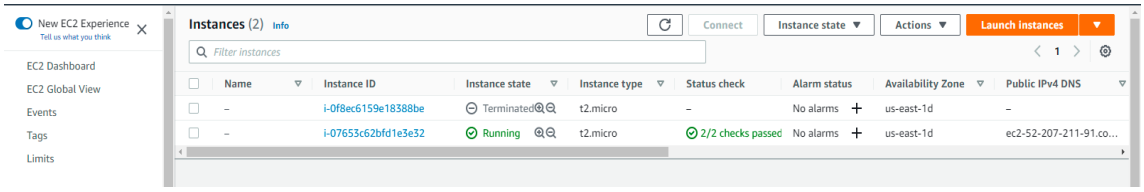
Step 7 : Click Next until you reach the “Configure Security Group” page.



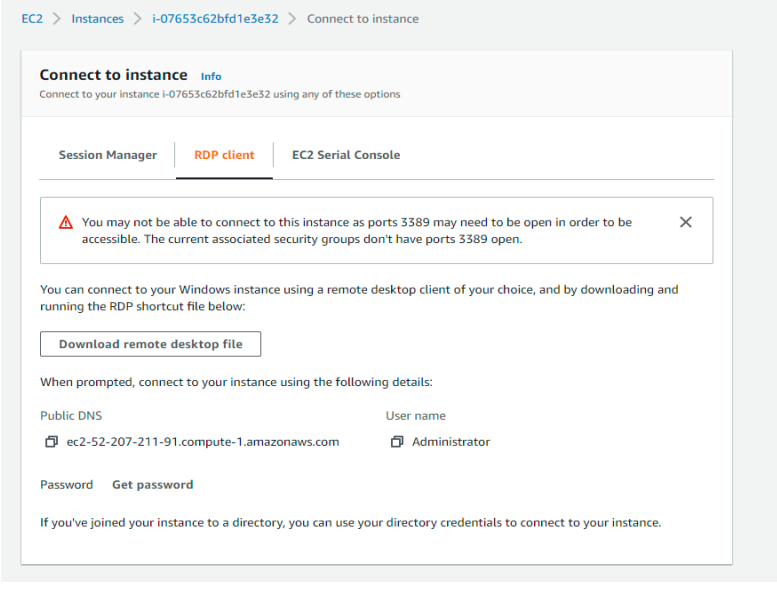
Step 8 : Click “Add Rule” and add HTTP and HTTPS protocols.



Step 9 : Wait until the status is “2/2 check passed” and select the instance and click connect



Step 10 : Now using the given details and using Remote desktop application connect to the windows server and host the webpage



5.2. Module Description

There are various modules of the Autonomous Library Management System. Each module has an interface within which a variety of functions can be enabled. The system controls access to the modules using access rights.

a) Cataloging Module

The librarian can use this module to capture book details in the most appropriate way, which enables easy retrieval of information concerning the book. Within this module, the system captures book details, such as the relevant authors, publishers, categories, and sections within the library.

b) Reservation Module

This module allows members to place reservations and check the status of their reservations. The librarian can respond to reservations.

c) User Management Module

Librarians can create new users using this module. They can also update the user's profile information.

d) Online Public Access Catalogue (OPAC) Module

With this module, members have access to books, wherever they are located. A number of different search criteria is available to them, such as searching books by their category, their author, their publisher, or even by their title.

5.3. Software Description

This web based software is designed using HTML, CSS, JavaScript, XAMPP, MySQL, uses AWS for cloud services and cloud server to host the website. Autonomous Library Management System is considered as SaaS (Software as a Service) as it is a cloud provider application that is available for users to use over the internet.

5.4. Sample Code

5.4.1. Login.php

```
<?php
session_start();
error_reporting(0);
include('includes/config.php');
if($_SESSION['login']!='')
{
$_SESSION['login']='';
}
if(isset($_POST['login']))
{
$email=$_POST['emailid'];
$password=md5($_POST['password']);
$sql ="SELECT EmailId,Password,StudentId,Status FROM tblstudents WHERE
EmailId=:email and Password=:password";
$query= $dbh -> prepare($sql);
$query-> bindParam(':email', $email, PDO::PARAM_STR);
$query-> bindParam(':password', $password, PDO::PARAM_STR);
$query-> execute();
$results=$query->fetchAll(PDO::FETCH_OBJ);
if($query->rowCount() > 0)
{
 foreach ($results as $result) {
 $_SESSION['stdid']=$result->StudentId;
if($result->Status==1)
{
$_SESSION['login']=$_POST['emailid'];
echo "<script type='text/javascript'> document.location ='dashboard.php'; </script>";
} else {
echo "<script>alert('Your Account Has been blocked .Please contact
admin');</script>";
```

```
}
}
}
else
{
echo "<script>alert('Invalid Details');</script>";
}
}
?>
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
   <meta charset="utf-8" />
   <meta name="viewport" content="width=device-width, initial-scale=1,
maximum-scale=1" />
   <meta name="description" content="" />
   <meta name="author" content="" />
   <title>Autonomous Library Management System | </title>
   <!--   CORE STYLE  -->
   <link href="assets/css/bootstrap.css" rel="stylesheet" />
   <!-- FONT STYLE  -->
   <link href="assets/css/font-awesome.css" rel="stylesheet" />
   <!-- CUSTOM STYLE  -->
   <link href="assets/css/style.css" rel="stylesheet" />
   <!-- GOOGLE FONT -->
   <link href='http://fonts.googleapis.com/css?family=Open+Sans' rel='stylesheet'
type='text/css' />

</head>
<body>
   <!------MENU START-->
<?php include('includes/header.php');?>
<!-- MENU  END-->
<div class="content-wrapper">
<div class="container">
```

```
<div class="row pad-botm">
<div class="col-md-12">
<h4 class="header-line">USER LOGIN FORM</h4>
</div>
</div>

<!--LOGIN PANEL START-->
<div class="row">
<div class="col-md-6 col-sm-6 col-xs-12 col-md-offset-3" >
<div class="panel panel-info">
<div class="panel-heading">
 LOGIN FORM
</div>
<div class="panel-body">
<form role="form" method="post">

<div class="form-group">
<label>Enter Email id</label>
<input class="form-control" type="text" name="emailid" required
autocomplete="off" />
</div>
<div class="form-group">
<label>Password</label>
<input class="form-control" type="password" name="password" required
autocomplete="off"  />
<p class="help-block"><a href="user-forgot-password.php">Forgot
Password</a></p>
</div>

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

```
<!---LOGIN PABNEL END-->

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

   <script src="assets/js/jquery-1.10.2.js"></script>
   <!-- BOOTSTRAP SCRIPTS  -->
   <script src="assets/js/bootstrap.js"></script>
     <!-- CUSTOM SCRIPTS  -->
   <script src="assets/js/custom.js"></script>

</body>
</html>
```

5.4.2. Dashboard.php

```
<?php
session_start();
error_reporting(0);
include('includes/config.php');
if(strlen($_SESSION['login'])==0)
  {
header('location:index.php');
}
else{?>
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
   <meta charset="utf-8" />
   <meta name="viewport" content="width=device-width, initial-scale=1,
maximum-scale=1" />
   <meta name="description" content="" />
   <meta name="author" content="" />
   <title>Autonomous Library Management System | User Dash Board</title>
```

```
    <!--   CORE STYLE  -->
    <link href="assets/css/bootstrap.css" rel="stylesheet" />
    <!-- FONT STYLE  -->
    <link href="assets/css/font-awesome.css" rel="stylesheet" />
    <!-- CUSTOM STYLE  -->
    <link href="assets/css/style.css" rel="stylesheet" />
    <!-- GOOGLE FONT -->
    <link href='http://fonts.googleapis.com/css?family=Open+Sans' rel='stylesheet'
type='text/css' />

</head>
<body>
     <!------MENU  START-->
<?php include('includes/header.php');?>
<!-- MENU  END-->
    <div class="content-wrapper">
         <div class="container">
        <div class="row pad-botm">
            <div class="col-md-12">
                <h4 class="header-line">USER DASHBOARD</h4>
                        </div>
        </div>
            <div class="row">
                 <div class="col-md-3 col-sm-3 col-xs-6">
     <div class="alert alert-info back-widget-set text-center">
  <i class="fa fa-bars fa-5x"></i>
<?php
$sid=$_SESSION['stdid'];
$sql1 ="SELECT id from tblissuedbookdetails where StudentID=:sid";
$query1 = $dbh -> prepare($sql1);
$query1->bindParam(':sid',$sid,PDO::PARAM_STR);
$query1->execute();
$results1=$query1->fetchAll(PDO::FETCH_OBJ);
$issuedbooks=$query1->rowCount();
?>
```

```
<h3><?php echo htmlentities($issuedbooks);?> </h3>
                    Book Issued
                  </div>
                </div>

            <div class="col-md-3 col-sm-3 col-xs-6">
                <div class="alert alert-warning back-widget-set text-center">
                    <i class="fa fa-recycle fa-5x"></i>
<?php
$rsts=0;
$sql2 ="SELECT id from tblissuedbookdetails where StudentID=:sid and
RetrunStatus=:rsts";
$query2 = $dbh -> prepare($sql2);
$query2->bindParam(':sid',$sid,PDO::PARAM_STR);
$query2->bindParam(':rsts',$rsts,PDO::PARAM_STR);
$query2->execute();
$results2=$query2->fetchAll(PDO::FETCH_OBJ);
$returnedbooks=$query2->rowCount();
?>

                    <h3><?php echo htmlentities($returnedbooks);?></h3>
                  Books Not Returned Yet
                  </div>
                </div>
      </div>
   </div>
   </div>

   <!-- JAVASCRIPT FILES PLACED AT THE BOTTOM TO REDUCE THE
LOADING TIME  -->
   <!-- CORE JQUERY  -->
   <script src="assets/js/jquery-1.10.2.js"></script>
   <!-- SCRIPTS  -->
   <script src="assets/js/bootstrap.js"></script>
    <!-- CUSTOM SCRIPTS  -->
    <script src="assets/js/custom.js"></script>
```

```
</body>
</html>
<?php } ?>
```

5.4.1. Logout.php

```
<?php
session_start();
$_SESSION = array();
if (ini_get("session.use_cookies")) {
    $params = session_get_cookie_params();
    setcookie(session_name(), '', time() - 60*60,
        $params["path"], $params["domain"],
        $params["secure"], $params["httponly"]
    );
}
unset($_SESSION['login']);
session_destroy(); // destroy session
header("location:index.php");
?>
```

5.4.1. Manage-books.php

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

```
$id=$_GET['del'];
$sql = "delete from tblbooks  WHERE id=:id";
$query = $dbh->prepare($sql);
$query -> bindParam(':id',$id, PDO::PARAM_STR);
$query -> execute();
$_SESSION['delmsg']="Category deleted scuccessfully ";
header('location:manage-books.php');

}
   ?>
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
   <meta charset="utf-8" />
   <meta name="viewport" content="width=device-width, initial-scale=1,
maximum-scale=1" />
   <meta name="description" content="" />
   <meta name="author" content="" />
   <title>Online Library Management System | Manage Books</title>
   <!--   CORE STYLE  -->
   <link href="assets/css/bootstrap.css" rel="stylesheet" />
   <!-- FONT STYLE  -->
   <link href="assets/css/font-awesome.css" rel="stylesheet" />
   <!-- DATATABLE STYLE  -->
   <link href="assets/js/dataTables/dataTables.bootstrap.css" rel="stylesheet" />
   <!-- CUSTOM STYLE  -->
   <link href="assets/css/style.css" rel="stylesheet" />
   <!-- GOOGLE FONT -->
   <link href='http://fonts.googleapis.com/css?family=Open+Sans' rel='stylesheet'
type='text/css' />

</head>
<body>
     <!------MENU START-->
<?php include('includes/header.php');?>
<!-- MENU END-->
```

```
    <div class="content-wrapper">
         <div class="container">
        <div class="row pad-botm">
            <div class="col-md-12">
                <h4 class="header-line">Manage Books</h4>
    </div>
     <div class="row">
    <?php if($_SESSION['error']!="")
    {?>
<div class="col-md-6">
<div class="alert alert-danger" >
 <strong>Error :</strong>
 <?php echo htmlentities($_SESSION['error']);?>
<?php echo htmlentities($_SESSION['error']="");?>
</div>
</div>
<?php } ?>
<?php if($_SESSION['msg']!="")
{?>
<div class="col-md-6">
<div class="alert alert-success" >
 <strong>Success :</strong>
 <?php echo htmlentities($_SESSION['msg']);?>
<?php echo htmlentities($_SESSION['msg']="");?>
</div>
</div>
<?php } ?>
<?php if($_SESSION['updatemsg']!="")
{?>
<div class="col-md-6">
<div class="alert alert-success" >
 <strong>Success :</strong>
 <?php echo htmlentities($_SESSION['updatemsg']);?>
<?php echo htmlentities($_SESSION['updatemsg']="");?>
</div>
</div>
```

```
<?php } ?>

  <?php if($_SESSION['delmsg']!="")
   {?>
<div class="col-md-6">
<div class="alert alert-success" >
 <strong>Success :</strong>
 <?php echo htmlentities($_SESSION['delmsg']);?>
<?php echo htmlentities($_SESSION['delmsg']="");?>
</div>
</div>
<?php } ?>

</div>
      </div>
         <div class="row">
            <div class="col-md-12">
               <!-- Advanced Tables -->
               <div class="panel panel-default">
                  <div class="panel-heading">
                    Books Listing
                  </div>
                  <div class="panel-body">
                     <div class="table-responsive">
                        <table class="table table-striped table-bordered table-hover"
id="dataTables-example">
                           <thead>
                              <tr>
                                 <th>#</th>
                                 <th>Book Name</th>
                                 <th>Category</th>
                                 <th>Author</th>
                                 <th>ISBN</th>
                                 <th>Price</th>
                                 <th>Action</th>
                              </tr>
```

```
                        </thead>
                        <tbody>
<?php $sql = "SELECT tblbooks.BookName,tblcategory.CategoryName,tblauthors.AuthorName,tblbooks.ISBNNumber,tblbooks.BookPrice,tblbooks.id as bookid from  tblbooks join tblcategory on tblcategory.id=tblbooks.CatId join tblauthors on tblauthors.id=tblbooks.AuthorId";
$query = $dbh -> prepare($sql);
$query->execute();
$results=$query->fetchAll(PDO::FETCH_OBJ);
$cnt=1;
if($query->rowCount() > 0)
{
foreach($results as $result)
{          ?>
            <tr class="odd gradeX">
                                <td class="center"><?php echo htmlentities($cnt);?></td>
                                <td class="center"><?php echo htmlentities($result->BookName);?></td>
                                <td class="center"><?php echo htmlentities($result->CategoryName);?></td>
                                <td class="center"><?php echo htmlentities($result->AuthorName);?></td>
                                <td class="center"><?php echo htmlentities($result->ISBNNumber);?></td>
                                <td class="center"><?php echo htmlentities($result->BookPrice);?></td>
                                <td class="center">

                                <a href="edit-book.php?bookid=<?php echo htmlentities($result->bookid);?>"><button class="btn btn-primary"><i class="fa fa-edit "></i> Edit</button>
                                <a href="manage-books.php?del=<?php echo htmlentities($result->bookid);?>" onclick="return confirm('Are you sure you want to delete?');"" >  <button class="btn btn-danger"><i class="fa fa-pencil"></i> Delete</button>
                                </td>
```

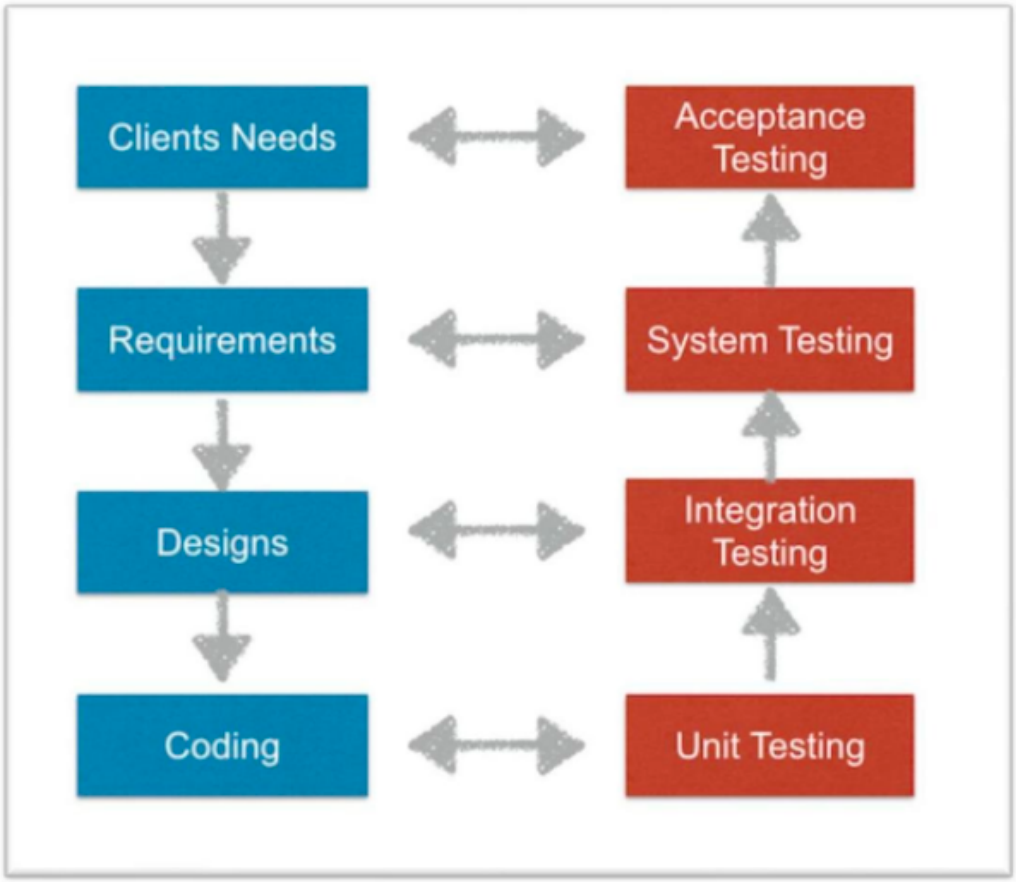
```
                                    </tr>
<?php $cnt=$cnt+1;}} ?>
                                </tbody>
                            </table>
                        </div>


                    </div>
                </div>
                <!--End Advanced Tables -->
            </div>
        </div>
    </div>
    </div>
        <!-- JAVASCRIPT FILES PLACED AT
                THE BOTTOM TO REDUCE THE LOADING TIME  -->
    <!-- CORE JQUERY  -->
    <script src="assets/js/jquery-1.10.2.js"></script>
    <!-- BOOTSTRAP SCRIPTS  -->
    <script src="assets/js/bootstrap.js"></script>
    <!-- DATATABLE SCRIPTS  -->
    <script src="assets/js/dataTables/jquery.dataTables.js"></script>
    <script src="assets/js/dataTables/dataTables.bootstrap.js"></script>
      <!-- CUSTOM SCRIPTS  -->
    <script src="assets/js/custom.js"></script>
</body>
</html>
<?php } ?>
```

6. SYSTEM TESTING

Testing is debugging of the program, it is one of the most critical aspects of computer programming. Without a program that works, the system would never produce an output for which it was designed. In testing, it is best to ask the user development department to identify all errors and bugs. Before the system goes live, it is tested to ensure that it operates accurately and efficiently.In the test data used for testing, it is not quantity, but quality of the data that counts.

Levels of Testing:

In order to discover the errors and bugs present in different phases we have the concept called levels of testing. The basic levels of testing are:



Unit testing:

During unit testing we test each module individually and then we integrate the module with the overall system. Unit testing focuses verification of the smallest unit of software design in the module, this testing is also known as module testing. Unit testing of a system is carried out during the stage of programming itself. In this each of the modules of the system are tested separately. The unit testing is successful only if each unit is found to work satisfactorily as regard to expected output from the unit. Each Module can be tested using the following two Strategies:

1. Black Box Testing
2. White Box Testing

Black Box Testing

A black box test is a technique for testing software functionality without looking at the code structure, implementation details, or knowledge of the internal path of the software.

White Box Testing

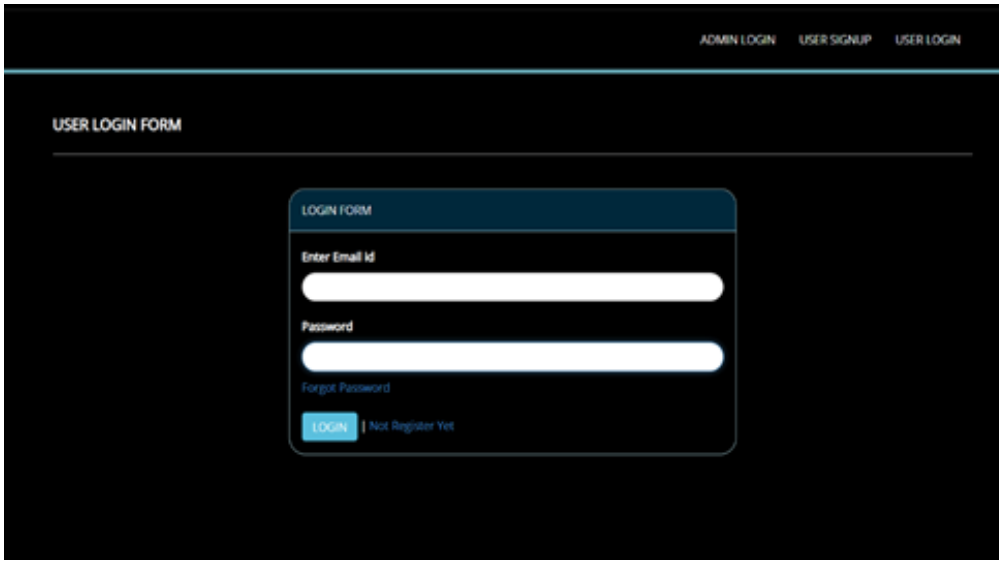
In White Box Testing we test the software of the internal coding and infrastructure. It focuses primarily on the flow of data through the application, improving security, and maximising design and usability.

System Testing:

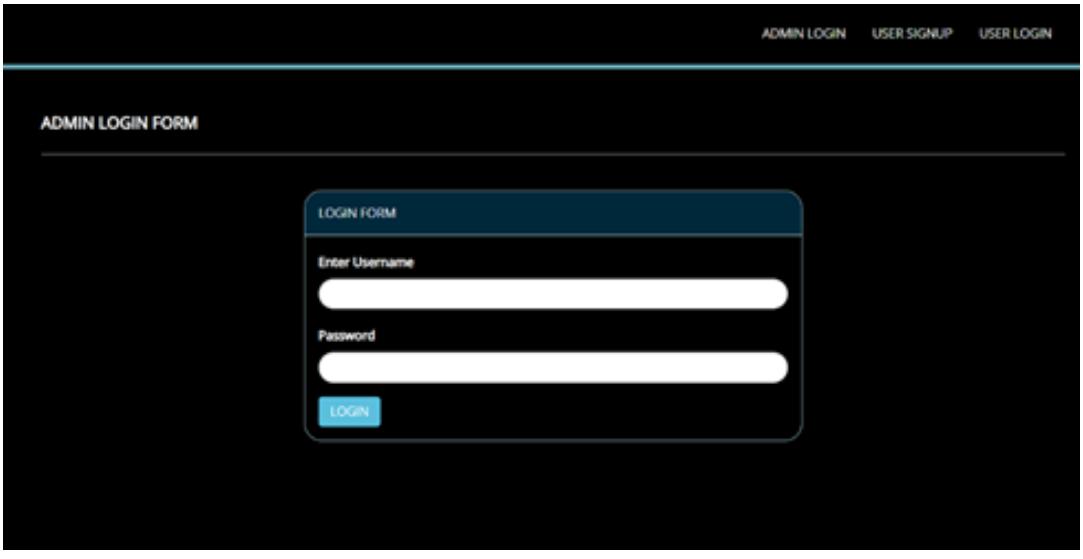
Once the individual unit testing is successfully completed, units are assembled and integrated to perform as a system analysis. The complete top to bottom testing, which includes complete modules upper level to lower level module, was carried out to verify if the entire system is performing as expected.

7. RESULTS SCREENSHOTS

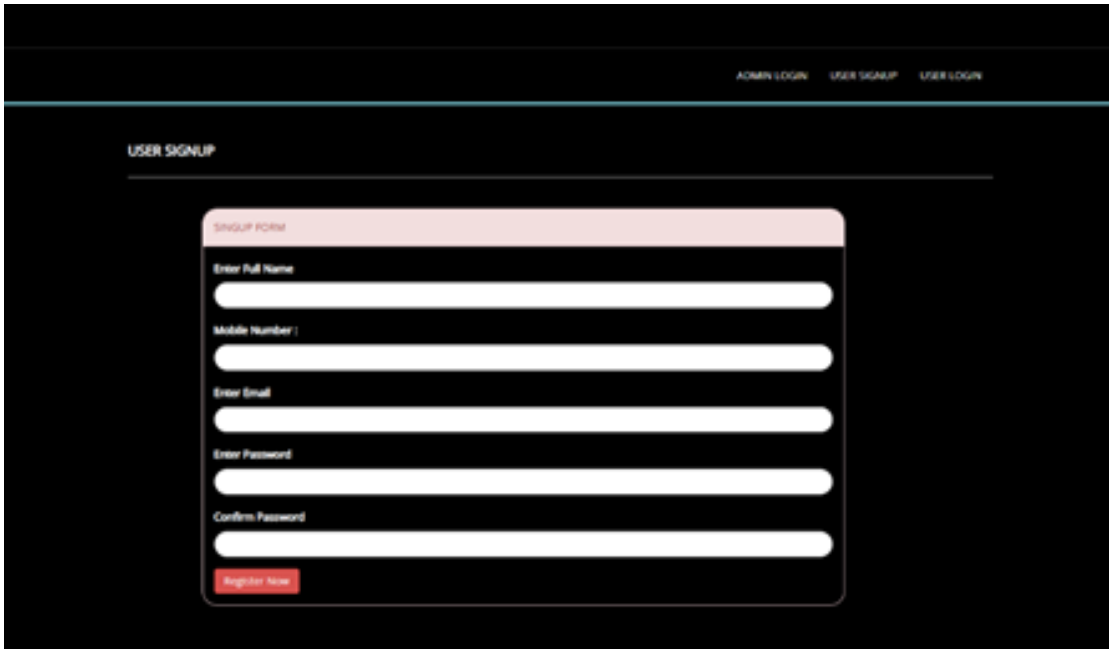
7.1. User Login



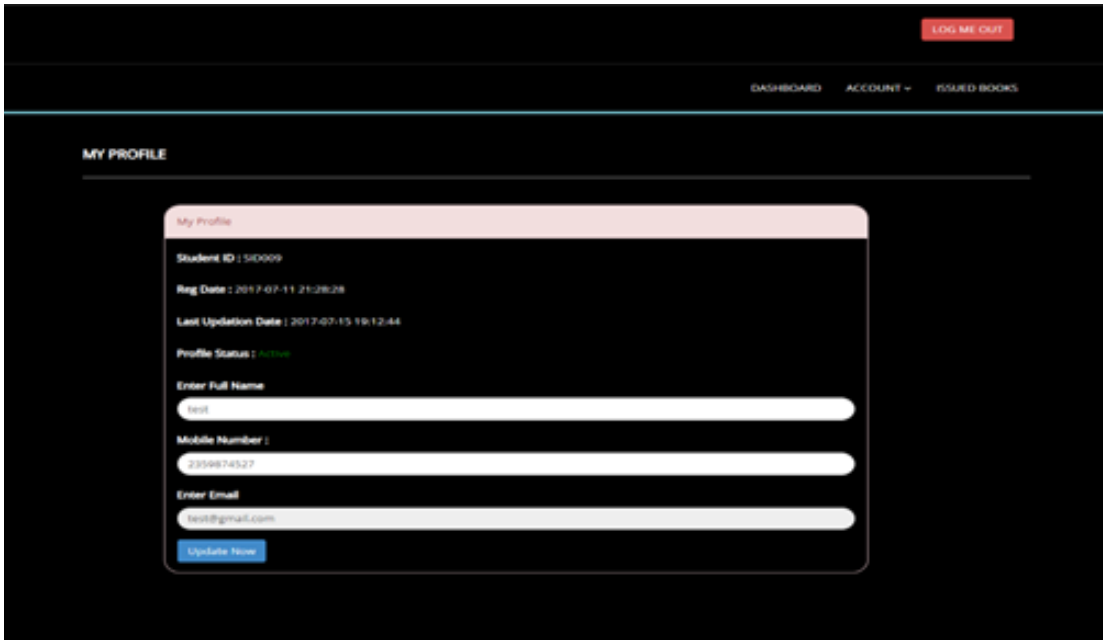
7.2. Admin Login



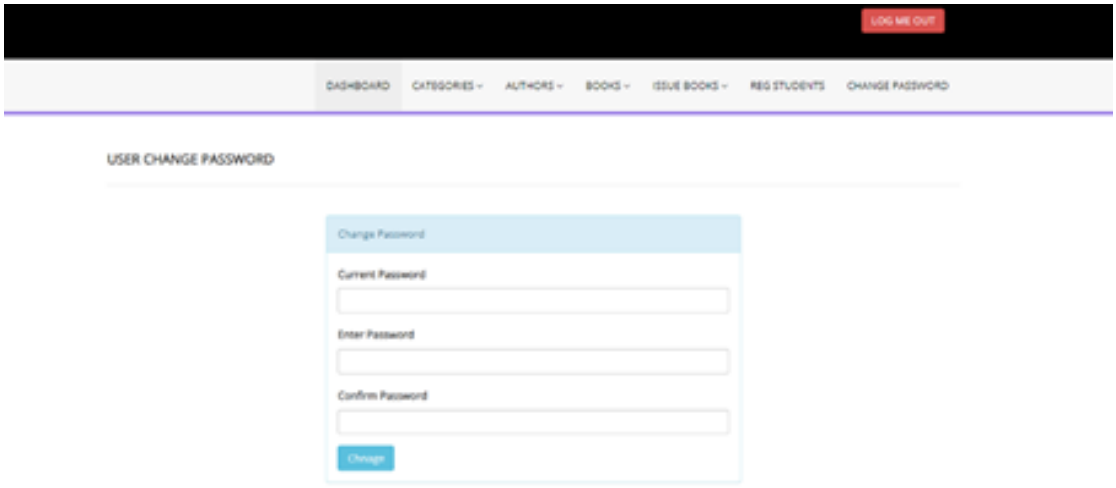
7.3. User Signup



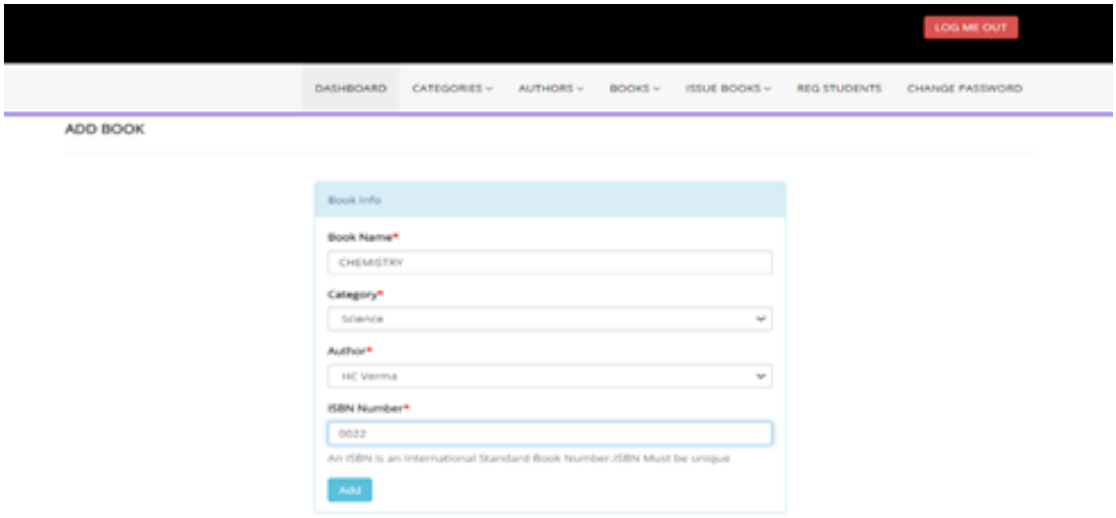
7.4. My Profile



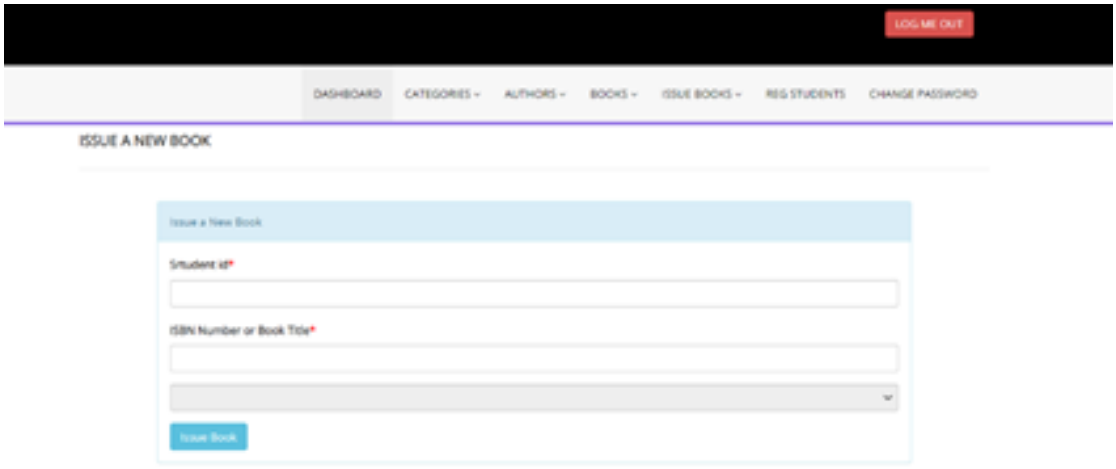
7.5. Change Password



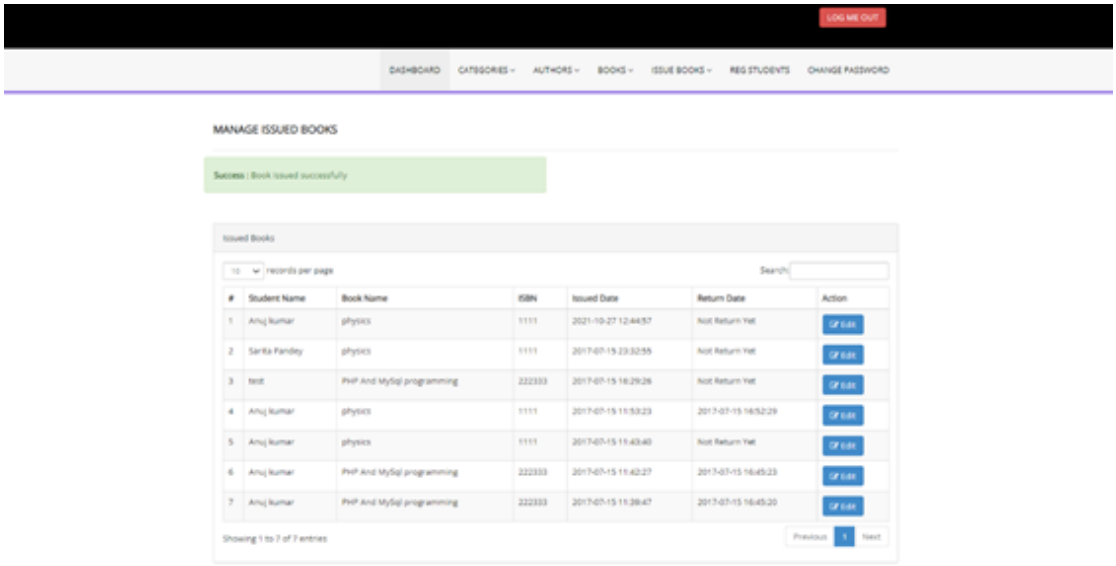
7.6. Add Book



7.7. Issue Book



7.8. Manage Books



8. CONCLUSION

In Conclusion the main aim of this project was to develop a library Management System that overcomes the drawbacks of the existing library Management Systems like:

- To prevent data loss using cloud services.
- Notify the borrowers the due date of the books they borrowed.
- Also an option to be notified when a particular book is available.
- Option to pre register a book.

We could fulfill the aim of this project by using AWS's servers to host the project on the cloud, and it is developed using knowledge in fields like HTML, CSS, JavaScript, and MySQL. This project includes member accounts which improve the user interaction and makes it easy for the librarian to maintain the library, it's books and also its members, and as this system run on cloud servers we don't have to worry about data loss and data corruption due to physical damage of the computer, hence this system can be accessed from anywhere using a web browser, it is easy to maintain, it is also fully managed, secure, the scalability, availability and durability of the data is very high.

8.1. FUTURE SCOPE

The future scope of the project is :

- To implement and integrate it into mobile applications and integrate features like notifications into devices like smart watches
- To design a system where in times of unavailability of internet the changes made in data are automatically changed in the cloud server as soon as the internet connection is established
- To implement Artificial intelligence and Machine learning algorithms to inform the librarian about when to arrange the books and what books to arrange in the book shelves based on the syllabus of the students and based on demand of the book.

9. BIBLIOGRAPHY

- Aggarwal, K.K., & Singh, Y. (2007). "*Software Engineering*" (3rd Ed.). New Age International Publishers.
- Roberts, P. (1999). "*Students Books reclaim priority status at library*". Marblehead Massachusetts.
- Pressman, Roger S. (2000) "*Software Engineering: A Practitioner's Approach*", (5th Ed.). McGraw-Hill.
- Alison Felstead. (2004). "*Library systems marketplace: a digest of current literature*" Program: electronic library and information systems.
- IEEE Standards 1016-1998, Recommended Practice for Software Design Descriptions
- IEEE Standards 830-1998, IEEE Recommended Practice for Software Requirements Specification
- "*HTML and CSS: Design and Build Websites*" – Jon Duckett
- "*JavaScript and JQuery: Interactive Front-End Web Development"* – Jon Duckett
- *"Responsive Web Design (Brief Books for People Who Make Websites)"* – Ethan Marcotte