**LIBRARY MANAGEMENT SYSTEM**

**BACHELOR OF SCIENCE (COMPUTER SCIENCE) BY**

# KHIZER ABDUL SALAM NAYAK

# PRN: 2020016400091264

**ROLL NO: 06**

Under the esteemed guidance of

# AFREEN FIROZ SHAIKH

Designation: HOD of CS/IT Department



Department Of Computer Science

# [M.S. College of Arts, Science, Commerce & BMS](http://www.hewscity.com/ms)

***(Affiliated To University of Mumbai)***

Mumbra, 400612 Maharashtra

# YEAR:2022-2023

**Project Certificate**



**Mumbra, 400612**

**Maharashtra**

**Year:2022-2023**

[**M.S. College of Arts, Science, Commerce & BMS**](http://www.hewscity.com/ms)

**This is to certify that the project entitled:**

**By Mr. / Ms:**

**Seat No:**

**in partial fulfillment of T.Y.B.Sc. (Computer Science) Semester V Degree 2022-2023 Examination had not been submitted for any other examination and does not form part of any other course undergone by the candidate.**

**It is further certified that he / she has completed all required phases of the project.**

**Internal Guide In charge**

**(Afreen Firoz Shaikh) (HOD of Computer Science Department)**

**Date: College seal Principal**

**(Saima S. Khan)**

**Signature of Examiner with Date**

**LIBRARY MANAGEMENT SYSTEM**

**PROFORMA FOR THE APPROVAL PROJECT PROPOSAL**

(Note: All entries of the perform approval should be filled up with appropriate and complete information.

Incomplete perform of approval in any request will be summarily rejected.)

**PRN NO:2020016400091264**

**ROLL NO: 06**

1. **Name of the Student: KHIZER ABDUL SALAM NAYAK**
2. **Title of the Project:**

**LIBRARY MANAGEMENT SYSTEM**

1. **Name of the Guide: AFREEN FIROZ SHAIKH**

**Designation: HOD of CS/IT Department**

**Signature of the Student Signature of the Guide**

**Date…………… Date……………**

**SIGNATURE OF THE COORDINATOR:**

**DATE………………………..**

# CERTIFICATE OF LIBRARY MANAGEMENT SYSTEM

**This is to Certify that Mr. KHIZER ABDUL SALAM NAYAK, Student of** [**M.S. College of Arts, Science, Commerce & BMS**](http://www.hewscity.com/ms) **has done sem V project for our services "LIBRARY MANAGEMENT SYSTEM”. We wish best of luck on completion of their project with us.**

**Their dedication and focus towards the project was good. We found it sincere and hardworking clauses of course of the project.**

**With best regards**

**Client Signature**

# ACKNOWLEDGEMENT

**I have taken efforts in this project. I However, it would not have been possible without the kind support and help of many individuals and guides. I would like to thank sincerely to all of them.**

**Miss. Afreen Firoz Shaikh took kind interest in checking the minute details of the project works and provide necessary information regarding the project and supported us to complete this project by guiding throughout the protect.**

**I am also very grateful to the “LIBRARY MANAGEMENT SYSTEM ” who gave me an opportunity to develop software for them and also for treating me with care. I would also like to thank my parents and friends who help me a lot in finalizing, this project within the limited time frame.**

**Finally, a note of thanks to the Department of Computer Science, both teaching and non-teaching staff for their co- operation towards students**.

# DECLARATION

**I Mr. KHIZER ABDUL SALAM NAYAK” of** [**M.S. College of Arts,**](http://www.hewscity.com/ms)[**Science, Commerce & BMS**](http://www.hewscity.com/ms) **, student of T.Y.B.Sc(Computer Science) Semester V hereby declare that we have completed the project on “LIBRARY MANAGEMENT SYSTEM” in the Academic year 2022-2023 is our original work and the information submitted is true and original to the best of our knowledge.**

**Place:Thane**

**Date: 20-10-2022**

**Signature of student**

**(KHIZER NAYAK)**

## TABLE OF CONTENTS

[ABSTRACT 8](#_TOC_250006)

1. INTRODUCTION 9
   1. [PROJECT AIMS AND OBJECTIVES 9](#_TOC_250005)
   2. [BACKGROUND OF PROJECT 10](#_TOC_250004)
   3. OPERATION ENVIRONMENT 11
2. SYSTEM ANALYSIS 12
   1. [SOFTWARE REQUIREMENT SPECIFICATION 12](#_TOC_250003)
   2. EXISTING VS PROPOSED 18
   3. SOFTWARE TOOL USED 19
3. SYSTEM DESIGN 20
   1. [TABLE DESIGN 23](#_TOC_250002)
   2. [DATA FLOW DIAGRAM’S 26](#_TOC_250001)
4. SYSTEM IMPLEMENTATION 27
   1. [MODULE DESCRIPTION 30](#_TOC_250000)
   2. SCREEN SHOTS 104
5. SYSTEM TESTING 110
   1. UNIT TESTING 113
   2. INTEGRATION TESTING 117
6. CONCLUSION & FUTURE SCOPE 119
7. REFERENCES 120

#### 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 i.e. 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.

**CHAPTER 1 INTRODUCTION**

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

#### 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.

#### 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 via Login/Signup 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. **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.

#### SOFTWARE REQUIREMENT SPECIFICATION

* + 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. Sometimes due to some human error there may be a loss of records.

* + - File damaged When a computerized system is not there file is always lost due to some accident like spilling of water by some member on file accidentally. Besides some natural disaster like floods or fires may also damage the files.
    - Difficult to search record

When there is no computerized system there is always a difficulty in searching of records if the records are large in number .

* + - Space consuming

After the number of records become large the space for physical storage of file and records also increases if no computerized system is implemented.

* + - Cost consuming

As there is no computerized system the to add each record paper will be needed which will increase the cost for the management of library.

#### 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

#### SYSTEM REQUIREMENTS

* + - 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 REQUIREMNTS

In implementing whole system it uses html in front end with C# 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.

* + - 1. FUNCTIONAL REQUIREMENTS

1. NORMAL USER
   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. 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. 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 Inventory part . we can search book based on book id, book name , publication or by author name and genre.

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. 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. 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 :- (This feature will be updated in upcoming bug fixes and patches)

-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

#### SOFTWARE AND HARDWARE REQUIREMENTS

This section describes the software and hardware requirements of the system

* + - 1. SOFTWARE REQUIREMENTS
         * Operating system- Windows 10 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,bootstrap,gridview for styling work and C# for sever side scripting.
      2. HARDWARE REQUIREMENTS
* Intel core i5 10th 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

#### EXISTING VS PROPOSED SYSTEM

* + 1. Existing system does not have any facility of teachers login or Search book via genre where as proposed system will have a facility of Search book as well as teacher’s login
    2. Existing system does not have a facility of online reservation of books whereas proposed system has a facility of online reservation of books
    3. 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.
    4. Existing system does not has any option of lectures notes uploaded by teachers whereas proposed system will have this facility
    5. 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
    6. Existing system does not has any facility for book request and suggestions where as in proposed system after logging in to their accounts student can request books as well as provide suggestions to improve library

#### SOFTWARE TOOLS USED

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

* + 1. Front end

The front end is designed using of html , css , Java script, bootstrap, gridView table,web fonts

* + - * + 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 <img>. 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.

* + - * + BOOTSTRAP – **BOOTSTRAP** Get started with Bootstrap, the world’s most popular framework for building responsive, mobile-first sites, with jsDelivr and a template starter page Bootstrap is the most popular HTML, CSS, and JavaScript framework for developing responsive, mobile-first websites.
        + C# - **C#** is a general-purpose, modern and object-oriented programming language pronounced as “C sharp”. It was developed by Microsoft led by Anders Hejlsberg and his team within the .Net initiative and was approved by the European Computer Manufacturers Association (ECMA) and International Standards Organization (ISO). C# is among the languages for Common language legislature and the current version of C# is version 7.2. C# is a lot similar to Java syntactically and is easy for the users who have knowledge of C, C++ or Java.

A bit about .Net Framework.Net applications are multi-platform applications and framework can be used from languages like C++, C#, Visual Basic, COBOL etc. It is designed in a manner so that other languages can use it.

* + 1. 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

#### TABLE DESIGN

VARIOUS TABELS TO MAINTAIN INFORMATION

* + - BOOK TABLE FOR KEEPING TRACK OF BOOKS

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Field** | **Data type** | **Default** | **Key** | **Extra** |
| BOOK ID | INT(10) | Not Null | Primary | Auto increment |
| Book Name | VARCHAR(255) | Null |  |  |
| Author Name | 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** |
| Member Id | INT(10) | NOT NULL | Primary key |  |
| Full Name | VARCHAR(50) | NULL |  |  |
| Email | VARCHAR(50) | NULL |  |  |
| Password | VARCHAR(50) | NULL |  |  |
| Account Status | VARCHAR(50) | NULL |  |  |
| Contact no | VARCHAR(50) | NULL |  |  |
| city | VARCHAR(50) | NULL |  |  |

* BOOK ISSUE TO A STUDENT/MEMBER

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Field** | **Data Type** | **Default** | **Key** | **Extra** |
| MEMBER ID | INT(10) | NOT NULL | Primary key | Auto increment |
| MEMBER NAME | VARCHAR(50) | NULL |  |  |
| BOOK ID | VARCHAR(50) | NULL |  |  |
| BOOK NAME | VARCHAR(50) | NULL |  |  |
| ISSUE DATE | VARCHAR(50) | NULL |  |  |
| DUE DATE | VARCHAR(50) | NULL |  |  |

* Book Details

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Field** | **Data Type** | **Default** | **Key** | **Extra** |
| BOOK ID | INT(11) | NOT NULL | Primary key | References book |
| GENRE | VARCHAR(50) | NULL |  | References Student |
| AUTHOR NAME | VARCHAR(50) | NULL |  |  |
| PUBLISHER DATE | Varchar(50) | NULL |  |  |

* STUDENT LOGIN TABLE

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Field** | **Data type** | **Default** | **Key** | **Extra** |
| Log Id | INT(10) | NOT NULL |  | References Student |
| Username | VARCHAR(50) | NULL |  |  |
| Password | VARCHAR(50) | NULL |  |  |

### ADMINISTRATOR TABLE:-

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Field** | **Data type** | **Default** | **Key** | **Extra** |
| UserName | Varchar(255) | NULL |  |  |
| Password | Date(yyyy/mm/dd) | NULL |  |  |

* TEACHER LOGIN TABLE

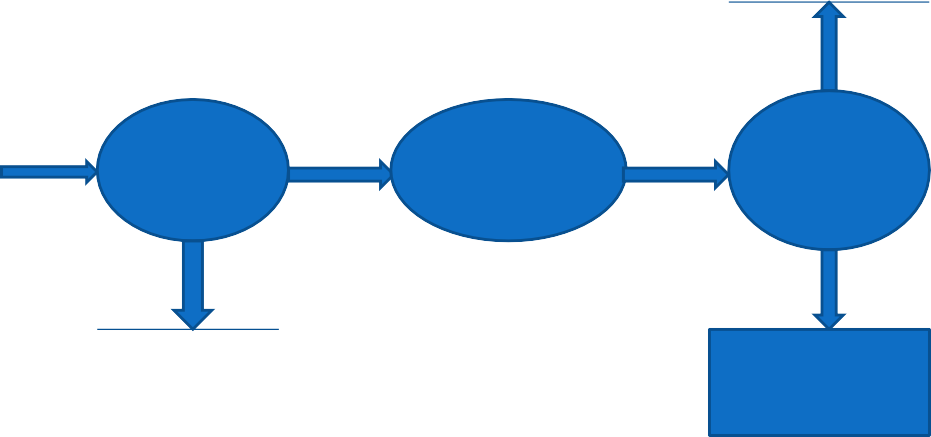
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Field** | **Data Type** | **Default** | **Key** | **Extra** |
| Login Id | INT(10) | NOT NULL | Primar key | References teacher |
| Username | VARCHAR(255) | NULL |  |  |
| Password | VARCHAR(255) | NULL |  |  |

#### DATA FLOW DIAGRAMS

DATA FLOW DIAGRAM FOR TEACHER LOGIN

TEACHER

DATABASE



ENTER

URL

ENTER

USERNAME

&

PASSWORD

CHECK

VALIDITY

IF VALID

WEB SERVER

TEACHER

LOGIN PAGE

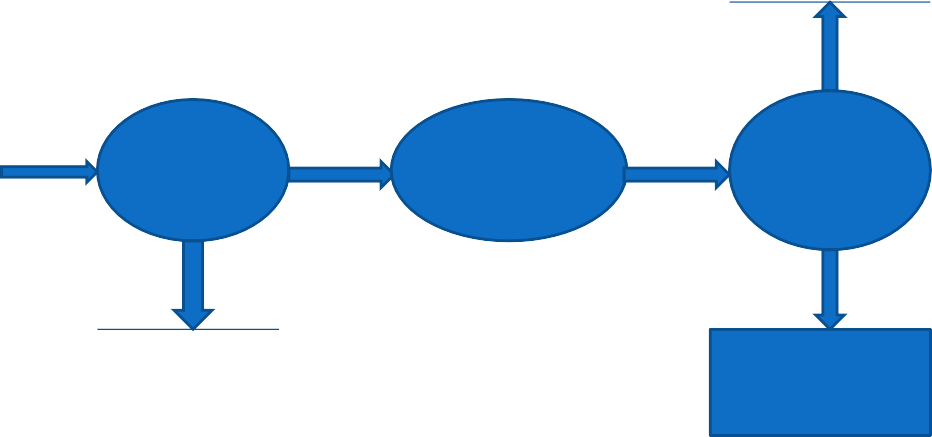
TEACHER

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

STUDENT

DATABASE



ENTER

URL

ENTER

USERNAME

&

PASSWORD

CHECK

VALIDITY

IF VALID

WEB SERVER

STUDENT

LOGIN PAGE

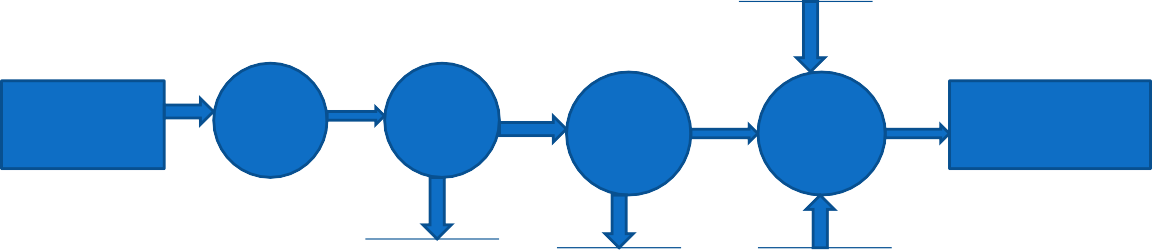
STUDENT

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

LOGIN

DATABASE



STUDENT

LOGIN

PAGE

SELECT

BOOK

ISSUE

ENTER

BOOK

DETAIL

IF

AVBL

SELECT

BOOK ISSUE

IF NO.

OF

BOOK

ISSUED LESS THAN

3

UPDATE

NO. OF

COPIES

MESSAGE

“BOOK

RESERVED”

BOOK

DATABASE

LOGIN

TABLE

BOOK

DATABASE

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

SELECT

ADVANCE SEARCH

ENTER

BOOK DETAIL

IF

FOUND

BOOK

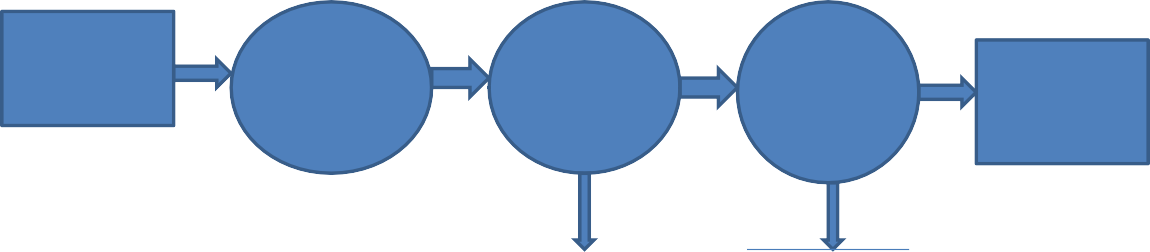
DETAILS

HOME

BOOK DATABASE

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



HOME

SELECT

ACCOUNT CREATION

ENTER

STUDENT DETAIL

ADMIN

VERIFY STUDEN T

MESSAGE

SENT

TO STUDENT

IF ALL REG.

PARTIALLY

IS

FILLED

IF VALID

ADMIN

DATABASE

STUDENT

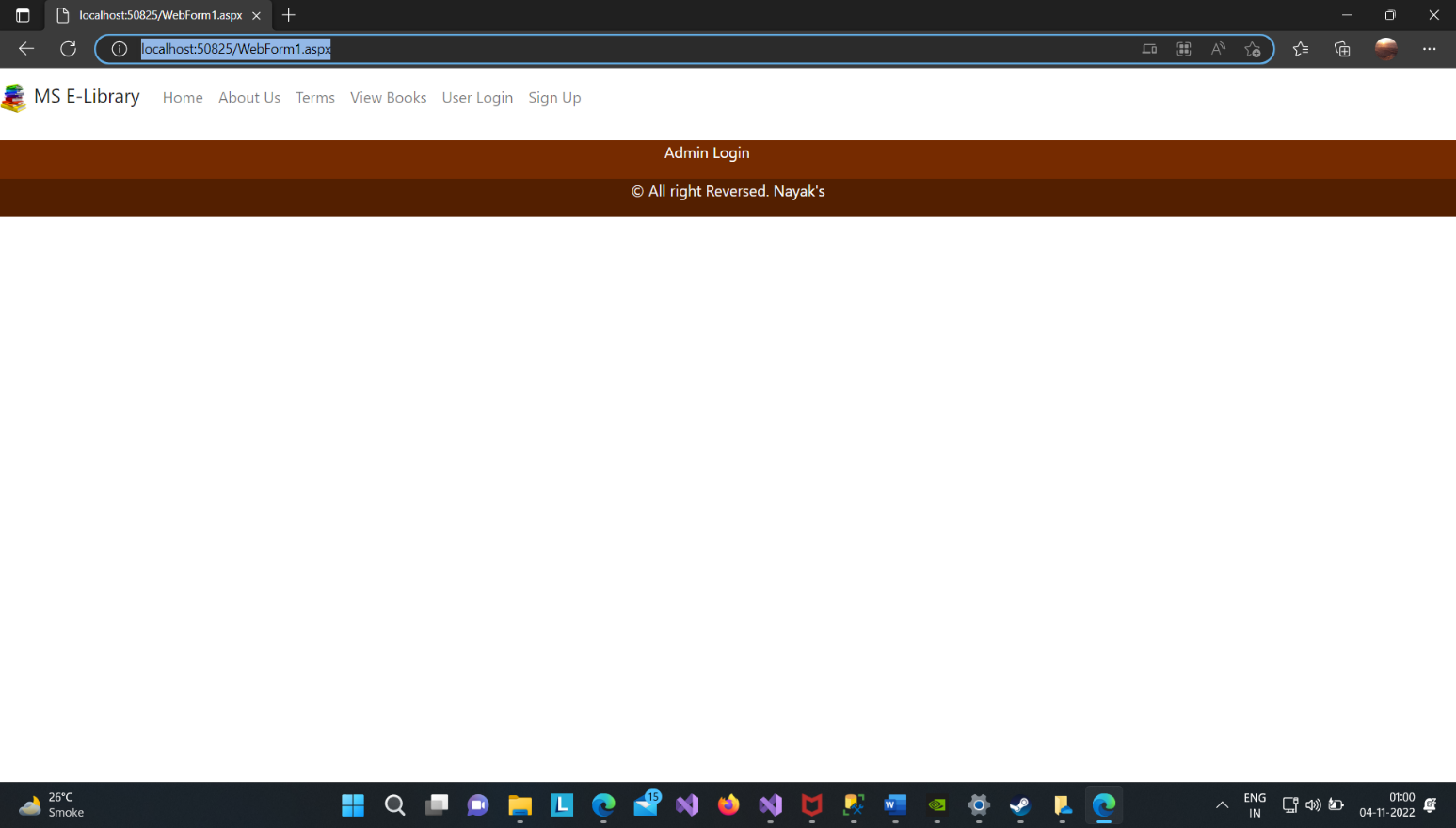
DATABASE

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**

****

#### MODULE DESCRIPTION

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

#### Admin Module

BOOK ADDITION

REPORT GENERATION

TEACHER REGISTRATION

ADMIN

STUDENT VALIDATION

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

**4.1.1.1 Code for Admin module in Visual Studio 2022**

* + - 1. Code for entering admin username and password :-

<%@ Page Title="" Language="C#" MasterPageFile="~/Site1.Master" AutoEventWireup="true" CodeBehind="Adminlogin.aspx.cs" Inherits="E\_library.Adminlogin" %>

<asp:Content ID="Content1" ContentPlaceHolderID="head" runat="server">

</asp:Content>

<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1" runat="server">

<!--Author: W3layouts

Author URL: http://w3layouts.com

License: Creative Commons Attribution 3.0 Unported

License URL: http://creativecommons.org/licenses/by/3.0/

-->

<!DOCTYPE HTML>

<html>

<head>

<title>ADMIN LOGIN PAGE</title>

<!-- Meta tag Keywords -->

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

<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />

<meta name="keywords" content="Flashy Login Form Responsive Widget,Login form widgets, Sign up Web forms , Login signup Responsive web form,Flat Pricing table,Flat Drop downs,Registration Forms,News letter Forms,Elements" />

<script type="application/x-javascript"> addEventListener("load", function() { setTimeout(hideURLbar, 0); }, false);

function hideURLbar(){ window.scrollTo(0,1); } </script>

<!-- Meta tag Keywords -->

<!-- css files -->

<!-- <link rel="stylesheet" href="css/style.css" type="text/css" media="all" /> -->

<link href="Content/Admin\_login/style.css" rel="stylesheet" type="text/css" media="all" />

<!-- Style-CSS -->

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

<link href="Content/Admin\_login/font-awesome.css" rel="stylesheet" />

<!-- Font-Awesome-Icons-CSS -->

<!-- //css files -->

<!-- online-fonts -->

<link href="//fonts.googleapis.com/css?family=Roboto:100,100i,300,300i,400,400i,500,500i,700,700i,900,900i&amp;subset=cyrillic,cyrillic-ext,greek,greek-ext,latin-ext,vietnamese" rel="stylesheet">

<!-- //online-fonts -->

</head>

<body>

<!-- main -->

<div data-vide-bg="Images/video/animation.mp4">

<div class="center-container">

<!--header-->

<div class="header-w3l">

<h1 style="font-family:Cambria, Cochin, Georgia, Times, Times New Roman, serif">ADMIN LOGIN </h1>

</div>

<!--//header-->

<div class="main-content-agile">

<div class="left-w3l-mk">

<img src="Images/adminuser.png" alt="Lib\_Login ">

<h3>Hey! SUPER USER </h3>

<!-- <div class="social-icon">

<a href="www.twitter.com" class="social-button twitter"><i class="fa fa-twitter"></i></a>

<a href="www.facebook.com" class="social-button facebook"><i class="fa fa-facebook"></i></a>

<a href="www.google.com" class="social-button google"><i class="fa fa-google-plus"></i></a>

</div> -->

</div>

<div class="sub-main-w3">

<div class="wthree-pro">

<h2><span>Connect</span> to Your Account</h2>

</div>

<form action="#" method="post">

<asp:TextBox ID="TextBox1" runat="server" placeholder="SUPER USER ID" name="Name" class="user" type="text" required > </asp:TextBox>

<span class="icon1"><i class="fa fa-user" aria-hidden="true"></i></span>

<span class="icon1"><i class="fa fa-user" aria-hidden="true"></i></span><br><br>

<asp:TextBox ID="TextBox2" runat="server" placeholder="Password" name="Password" class="pass" type="password" required></asp:TextBox>

<span class="icon2"><i class="fa fa-unlock" aria-hidden="true"></i></span><br>

<div class="demo5">

<div class="switch demo3">

<input type="checkbox">

<label><i></i></label>

</div>

</div>

<div class="sub-w3l">

<h6><a href="#">Stay Signed In</a></h6>

<div class="right-w3l">

<asp:Button ID="Button1" runat="server" type="submit" Text=Login OnClick="Button1\_Click" />

</div>

</div>

<div class="sub-w3l">

<div class="right-w3l">

<asp:LinkButton ID="Button2" type="link" href="Userlogin.aspx" runat="server" >USER LOGIN</asp:LinkButton>

</div>

</div>

</form>

</div>

<div class="clear"></div>

</div>

<!--//main-->

<!--footer-->

<div class="footer" hidden >

<p>&copy; 2017 Flashy Login Form. All rights reserved | Design by <a href="http://w3layouts.com">W3layouts</a></p>

</div>

<!--//footer-->

</div>

</div>

<!-- js -->

<!-- <script type="text/javascript" src="js/jquery-2.1.4.min.js"></script> -->

<script type="text/javascript" src="Scripts/admin\_login/jquery-2.1.4.min.js"></script>

<!-- //js -->

<!-- <script src="js/jquery.vide.min.js"></script> -->

<script src="Scripts/admin\_login/jquery.vide.min.js"></script>

</body>

</html>

</asp:Content>

* + - 1. **Code for checking of admin username and password information**

using System;

using System.Collections.Generic;

using System.Configuration;

using System.Data;

using System.Data.SqlClient;

using System.Linq;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

namespace E\_library

{

public partial class Adminlogin : System.Web.UI.Page

{

string strcon = ConfigurationManager.ConnectionStrings["con"].ConnectionString;

protected void Page\_Load(object sender, EventArgs e)

{

}

//login button click event

protected void Button1\_Click(object sender, EventArgs e)

{

try

{

SqlConnection con = new SqlConnection(strcon);

if (con.State == ConnectionState.Closed)

{

con.Open();

}

SqlCommand cmd = new SqlCommand("Select \* from admin\_login\_tbl where username ='" + TextBox1.Text.Trim() + "' AND password ='" + TextBox2.Text.Trim() + "' ", con);

SqlDataReader dr = cmd.ExecuteReader();

if (dr.HasRows)

{

while (dr.Read())

{

Response.Write("<script>alert('" + dr.GetValue(0).ToString() + "')</script>");

Session["username"] = dr.GetValue(0).ToString();

Session["fullname"] = dr.GetValue(2).ToString();

Session["role"] = "admin";

//Session["status"] = dr.GetValue(10).ToString();

}

Response.Redirect("homepage.aspx");

}

else

{

Response.Write("<script>alert('You cannot excess this area ');</script>");

}

}

catch (Exception ex)

{

Response.Write("<script>alert('"+ex.Message+"')</script>");

}

}

}

}

* + - 1. **Code for student signup/login validation**

1. using System;
2. using System.Collections.Generic;
3. using System.Configuration;
4. using System.Data;
5. using System.Data.SqlClient;
6. using System.Linq;
7. using System.Web;
8. using System.Web.UI;
9. using System.Web.UI.WebControls;
10. namespace E\_library
11. {
12. public partial class Signup : System.Web.UI.Page
13. {
14. string strcon = ConfigurationManager.ConnectionStrings["con"].ConnectionString;
15. protected void Page\_Load(object sender, EventArgs e)
16. {
17. }
18. //signup button is click
19. protected void Button1\_Click(object sender, EventArgs e)
20. {
21. if (checkMemberExists())
22. {
23. Response.Write("<script>alert('Member Already Exist with this Member ID, try other ID');</script>");
24. }
25. else
26. {
27. signUpNewMember();
28. }
29. // user defined method
30. bool checkMemberExists()
31. {
32. try
33. {
34. SqlConnection con = new SqlConnection(strcon);
35. if (con.State == ConnectionState.Closed)
36. {
37. con.Open();
38. }
39. SqlCommand cmd = new SqlCommand("SELECT \* from member\_master\_tbl where member\_id='" + TextBox8.Text.Trim() + "'", con);
40. SqlDataAdapter da = new SqlDataAdapter(cmd);
41. DataTable dt = new DataTable();
42. da.Fill(dt);
43. if (dt.Rows.Count >= 1)
44. {
45. return true;
46. }
47. else
48. {
49. return false;
50. }
51. }
52. catch (Exception ex)
53. {
54. Response.Write("<script>alert('" + ex.Message + "');</script>");
55. return false;
56. }
57. }
58. void signUpNewMember()
59. {
60. // Response.Write("<script>alert('Testing');</script>");
61. try
62. {
63. SqlConnection con = new SqlConnection(strcon);
64. if (con.State == ConnectionState.Closed)
65. {
66. con.Open();
67. }
68. SqlCommand cmd = new SqlCommand("INSERT INTO member\_master\_tbl(full\_name,dob,contact\_no,email,state,city,pincode,full\_address,member\_id,password,account\_status) values(@full\_name,@dob,@contact\_no,@email,@state,@city,@pincode,@full\_address,@member\_id,@password,@account\_status)", con);
69. cmd.Parameters.AddWithValue("@full\_name", TextBox1.Text.Trim());
70. cmd.Parameters.AddWithValue("@dob", TextBox2.Text.Trim());
71. cmd.Parameters.AddWithValue("@contact\_no", TextBox3.Text.Trim());
72. cmd.Parameters.AddWithValue("@email", TextBox4.Text.Trim());
73. cmd.Parameters.AddWithValue("@state", DropDownList.SelectedItem.Value);
74. cmd.Parameters.AddWithValue("@city", TextBox5.Text.Trim());
75. cmd.Parameters.AddWithValue("@pincode", TextBox6.Text.Trim());
76. cmd.Parameters.AddWithValue("@full\_address", TextBox7.Text.Trim());
77. cmd.Parameters.AddWithValue("@member\_id", TextBox8.Text.Trim());
78. cmd.Parameters.AddWithValue("@password", TextBox9.Text.Trim());
79. cmd.Parameters.AddWithValue("@account\_status", "pending");
80. cmd.ExecuteNonQuery();
81. con.Close();
82. Response.Write("<script>alert('Sign Up Successful.Go to User Login to Login');</script>");
83. }
84. catch (Exception ex)
85. {
86. Response.Write("<script>alert('" + ex.Message + "');</script>");
87. }
88. }
89. }
90. }
91. }

#### Code For Student Login Credentials:-

1. using System;
2. using System.Collections.Generic;
3. using System.Configuration;
4. using System.Data;
5. using System.Data.SqlClient;
6. using System.Linq;
7. using System.Web;
8. using System.Web.UI;
9. using System.Web.UI.WebControls;
10. namespace E\_library
11. {
12. public partial class Userlogin : System.Web.UI.Page
13. {
14. string strcon = ConfigurationManager.ConnectionStrings["con"].ConnectionString;
15. protected void Page\_Load(object sender, EventArgs e)
16. {
17. }
18. //login button click event
19. protected void Button1\_Click1(object sender, EventArgs e)
20. {
21. try
22. {
23. SqlConnection con = new SqlConnection(strcon);
24. if (con.State == ConnectionState.Closed)
25. {
26. con.Open();
27. }
28. SqlCommand cmd = new SqlCommand("Select \* from member\_master\_tbl where member\_id='" + TextBox1.Text.Trim() + "' AND password='" + TextBox2.Text.Trim() + "'", con);
29. SqlDataReader dr = cmd.ExecuteReader();
30. if (dr.HasRows)
31. {
32. while (dr.Read())
33. {
34. Response.Write("<script>alert('" + dr.GetValue(8).ToString() + "');</script>");
35. Session["username"] = dr.GetValue(8).ToString();
36. Session["fullname"] = dr.GetValue(0).ToString();
37. Session["role"] = "user";
38. Session["status"] = dr.GetValue(10).ToString();
39. }
40. Response.Redirect("homePage.aspx");
41. }
42. else
43. {
44. Response.Write("<script>alert('Invalid Username Or Password ');</script>");
45. }
46. }
47. catch (Exception ex)
48. {
49. Response.Write("<script>alert('" + ex.Message + " ');</script>");
50. }
51. }
52. protected void ClickButton2\_Click(object sender, EventArgs e)
53. {
54. Response.Redirect("HomePage.aspx");
55. }
56. }

#### }

#### UserView Profile design :-

#### <%@ Page Title="" Language="C#" MasterPageFile="~/Site1.Master" AutoEventWireup="true" CodeBehind="Userviewprofile.aspx.cs" Inherits="E\_library.Userviewprofile" %>

#### <asp:Content ID="Content1" ContentPlaceHolderID="head" runat="server">

#### <link href="Content/Userviewprofile/css/StyleSheet1.css" rel="stylesheet" />

#### </asp:Content>

#### <asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1" runat="server">

#### <div class="container-fluid">

#### <div class="row">

#### <div class="col-md-5">

#### <div class="card">

#### <div class="card-body">

#### <div class="row">

#### <div class="col">

#### <center>

#### <img src="Images/generaluser.png" width="100px" />

#### </center>

#### </div>

#### </div>

#### <div class="row">

#### <div class="col">

#### <center>

#### <h4>Your Profile</h4>

#### <span>Account Status - </span>

#### <asp:Label class="badge badge-pill badge-info" ID="Label1" runat="server" Text="Your status"></asp:Label>

#### </center>

#### </div>

#### </div>

#### <div class="row">

#### <div class="col">

#### <hr>

#### </div>

#### </div>

#### <div class="row">

#### <div class="col-md-6">

#### <label>Full Name</label>

#### <div class="form-group">

#### <asp:TextBox CssClass="form-control" ID="TextBox1" runat="server" placeholder="Full Name"></asp:TextBox>

#### </div>

#### </div>

#### <div class="col-md-6">

#### <label>Date of Birth</label>

#### <div class="form-group">

#### <asp:TextBox CssClass="form-control" ID="TextBox2" runat="server" placeholder="Password" TextMode="Date"></asp:TextBox>

#### </div>

#### </div>

#### </div>

#### <div class="row">

#### <div class="col-md-6">

#### <label>Contact No</label>

#### <div class="form-group">

#### <asp:TextBox CssClass="form-control" ID="TextBox3" runat="server" placeholder="Contact No" TextMode="Number"></asp:TextBox>

#### </div>

#### </div>

#### <div class="col-md-6">

#### <label>Email ID</label>

#### <div class="form-group">

#### <asp:TextBox CssClass="form-control" ID="TextBox4" runat="server" placeholder="Email ID" TextMode="Email"></asp:TextBox>

#### </div>

#### </div>

#### </div>

#### <div class="row">

#### <div class="col-md-4">

#### <label>State</label>

#### <div class="form-group">

#### <asp:DropDownList class="form-control" ID="DropDownList1" runat="server">

#### <asp:ListItem Text="Select" Value="select" />

#### <asp:ListItem Text="Andhra Pradesh" Value="Andhra Pradesh" />

#### <asp:ListItem Text="Arunachal Pradesh" Value="Arunachal Pradesh" />

#### <asp:ListItem Text="Assam" Value="Assam" />

#### <asp:ListItem Text="Bihar" Value="Bihar" />

#### <asp:ListItem Text="Chhattisgarh" Value="Chhattisgarh" />

#### <asp:ListItem Text="Rajasthan" Value="Rajasthan" />

#### <asp:ListItem Text="Goa" Value="Goa" />

#### <asp:ListItem Text="Gujarat" Value="Gujarat" />

#### <asp:ListItem Text="Haryana" Value="Haryana" />

#### <asp:ListItem Text="Himachal Pradesh" Value="Himachal Pradesh" />

#### <asp:ListItem Text="Jammu and Kashmir" Value="Jammu and Kashmir" />

#### <asp:ListItem Text="Jharkhand" Value="Jharkhand" />

#### <asp:ListItem Text="Karnataka" Value="Karnataka" />

#### <asp:ListItem Text="Kerala" Value="Kerala" />

#### <asp:ListItem Text="Madhya Pradesh" Value="Madhya Pradesh" />

#### <asp:ListItem Text="Maharashtra" Value="Maharashtra" />

#### <asp:ListItem Text="Manipur" Value="Manipur" />

#### <asp:ListItem Text="Meghalaya" Value="Meghalaya" />

#### <asp:ListItem Text="Mizoram" Value="Mizoram" />

#### <asp:ListItem Text="Nagaland" Value="Nagaland" />

#### <asp:ListItem Text="Odisha" Value="Odisha" />

#### <asp:ListItem Text="Punjab" Value="Punjab" />

#### <asp:ListItem Text="Rajasthan" Value="Rajasthan" />

#### <asp:ListItem Text="Sikkim" Value="Sikkim" />

#### <asp:ListItem Text="Tamil Nadu" Value="Tamil Nadu" />

#### <asp:ListItem Text="Telangana" Value="Telangana" />

#### <asp:ListItem Text="Tripura" Value="Tripura" />

#### <asp:ListItem Text="Uttar Pradesh" Value="Uttar Pradesh" />

#### <asp:ListItem Text="Uttarakhand" Value="Uttarakhand" />

#### <asp:ListItem Text="West Bengal" Value="West Bengal" />

#### </asp:DropDownList>

#### </div>

#### </div>

#### <div class="col-md-4">

#### <label>City</label>

#### <div class="form-group">

#### <asp:TextBox class="form-control" ID="TextBox6" runat="server" placeholder="City"></asp:TextBox>

#### </div>

#### </div>

#### <div class="col-md-4">

#### <label>Pincode</label>

#### <div class="form-group">

#### <asp:TextBox class="form-control" ID="TextBox7" runat="server" placeholder="Pincode" TextMode="Number"></asp:TextBox>

#### </div>

#### </div>

#### </div>

#### <div class="row">

#### <div class="col">

#### <label>Full Address</label>

#### <div class="form-group">

#### <asp:TextBox CssClass="form-control" ID="TextBox5" runat="server" placeholder="Full Address" TextMode="MultiLine" Rows="2"></asp:TextBox>

#### </div>

#### </div>

#### </div>

#### <div class="row">

#### <div class="col">

#### <center>

#### <span class="badge badge-pill badge-info">Login Credentials</span>

#### </center>

#### </div>

#### </div>

#### <div class="row">

#### <div class="col-md-4">

#### <label>User ID/Member ID</label>

#### <div class="form-group">

#### <asp:TextBox class="form-control" ID="TextBox8" runat="server" placeholder="User ID/Member ID" ReadOnly="True"></asp:TextBox>

#### </div>

#### </div>

#### <div class="col-md-4">

#### <label>Old Password</label>

#### <div class="form-group">

#### <asp:TextBox class="form-control" ID="TextBox9" runat="server" placeholder="Old Password" TextMode="Password" ReadOnly="True"></asp:TextBox>

#### </div>

#### </div>

#### <div class="col-md-4">

#### <label>New Password</label>

#### <div class="form-group">

#### <asp:TextBox class="form-control" ID="TextBox10" runat="server" placeholder="New Password" TextMode="Password"></asp:TextBox>

#### </div>

#### </div>

#### </div>

#### <div class="row">

#### <div class="col-8 mx-auto">

#### <center>

#### <br />

#### <div class="form-group">

#### <asp:Button class="btn btn-primary btn-block btn-lg" ID="Button1" runat="server" Text="Update" OnClick="Button1\_Click" />

#### </div>

#### </center>

#### </div>

#### </div>

#### </div>

#### </div>

#### <a href="homepage.aspx"> << Back to Home</a><br><br>

#### </div>

#### <div class="col-md-7">

#### <div class="card">

#### <div class="card-body">

#### <div class="row">

#### <div class="col">

#### <center>

#### <img src="Images/books1.png" width=" 100px "/>

#### </center>

#### </div>

#### </div>

#### <div class="row">

#### <div class="col">

#### <center>

#### <h4>Your Issued Books</h4>

#### <asp:Label class="badge badge-pill badge-info" ID="Label2" runat="server" Text="your books info"></asp:Label>

#### </center>

#### </div>

#### </div>

#### <div class="row">

#### <div class="col">

#### <hr>

#### </div>

#### </div>

#### <div class="row">

#### <div class="col">

#### <asp:GridView class="table table-striped table-bordered" ID="GridView1" runat="server" OnRowDataBound="GridView1\_RowDataBound"></asp:GridView>

#### </div>

#### </div>

#### </div>

#### </div>

#### </div>

#### </div>

#### </div>

#### </asp:Content>

#### UserView Profile Code:-

using System;

using System.Collections.Generic;

using System.Configuration;

using System.Data.SqlClient;

using System.Data;

using System.Linq;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

namespace E\_library

{

public partial class Userviewprofile : System.Web.UI.Page

{

string strcon = ConfigurationManager.ConnectionStrings["con"].ConnectionString;

protected void Page\_Load(object sender, EventArgs e)

{

try

{

if (Session["username"].ToString() == "" || Session["username"] == null)

{

Response.Write("<script>alert('Session Expired Login Again');</script>");

Response.Redirect("userlogin.aspx");

}

else

{

getUserBookData();

if (!Page.IsPostBack)

{

getUserPersonalDetails();

}

}

}

catch (Exception ex)

{

Response.Write("<script>alert('Session Expired Login Again');</script>");

Response.Redirect("userlogin.aspx");

}

}

// update button click

protected void Button1\_Click(object sender, EventArgs e)

{

if (Session["username"].ToString() == "" || Session["username"] == null)

{

Response.Write("<script>alert('Session Expired Login Again');</script>");

Response.Redirect("userlogin.aspx");

}

else

{

updateUserPersonalDetails();

}

}

// user defined function

void updateUserPersonalDetails()

{

string password = "";

if (TextBox10.Text.Trim() == "")

{

password = TextBox9.Text.Trim();

}

else

{

password = TextBox10.Text.Trim();

}

try

{

SqlConnection con = new SqlConnection(strcon);

if (con.State == ConnectionState.Closed)

{

con.Open();

}

SqlCommand cmd = new SqlCommand("update member\_master\_tbl set full\_name=@full\_name, dob=@dob, contact\_no=@contact\_no, email=@email, state=@state, city=@city, pincode=@pincode, full\_address=@full\_address, password=@password, account\_status=@account\_status WHERE member\_id='" + Session["username"].ToString().Trim() + "'", con);

cmd.Parameters.AddWithValue("@full\_name", TextBox1.Text.Trim());

cmd.Parameters.AddWithValue("@dob", TextBox2.Text.Trim());

cmd.Parameters.AddWithValue("@contact\_no", TextBox3.Text.Trim());

cmd.Parameters.AddWithValue("@email", TextBox4.Text.Trim());

cmd.Parameters.AddWithValue("@state", DropDownList1.SelectedItem.Value);

cmd.Parameters.AddWithValue("@city", TextBox6.Text.Trim());

cmd.Parameters.AddWithValue("@pincode", TextBox7.Text.Trim());

cmd.Parameters.AddWithValue("@full\_address", TextBox5.Text.Trim());

cmd.Parameters.AddWithValue("@password", password);

cmd.Parameters.AddWithValue("@account\_status", "pending");

int result = cmd.ExecuteNonQuery();

con.Close();

if (result > 0)

{

Response.Write("<script>alert('Your Details Updated Successfully');</script>");

getUserPersonalDetails();

getUserBookData();

}

else

{

Response.Write("<script>alert('Invaid entry');</script>");

}

}

catch (Exception ex)

{

Response.Write("<script>alert('" + ex.Message + "');</script>");

}

}

void getUserPersonalDetails()

{

try

{

SqlConnection con = new SqlConnection(strcon);

if (con.State == ConnectionState.Closed)

{

con.Open();

}

SqlCommand cmd = new SqlCommand("SELECT \* from member\_master\_tbl where member\_id='" + Session["username"].ToString() + "';", con);

SqlDataAdapter da = new SqlDataAdapter(cmd);

DataTable dt = new DataTable();

da.Fill(dt);

TextBox1.Text = dt.Rows[0]["full\_name"].ToString();

TextBox2.Text = dt.Rows[0]["dob"].ToString();

TextBox3.Text = dt.Rows[0]["contact\_no"].ToString();

TextBox4.Text = dt.Rows[0]["email"].ToString();

DropDownList1.SelectedValue = dt.Rows[0]["state"].ToString().Trim();

TextBox6.Text = dt.Rows[0]["city"].ToString();

TextBox7.Text = dt.Rows[0]["pincode"].ToString();

TextBox5.Text = dt.Rows[0]["full\_address"].ToString();

TextBox8.Text = dt.Rows[0]["member\_id"].ToString();

TextBox9.Text = dt.Rows[0]["password"].ToString();

Label1.Text = dt.Rows[0]["account\_status"].ToString().Trim();

if (dt.Rows[0]["account\_status"].ToString().Trim() == "active")

{

Label1.Attributes.Add("class", "badge badge-pill badge-success");

}

else if (dt.Rows[0]["account\_status"].ToString().Trim() == "pending")

{

Label1.Attributes.Add("class", "badge badge-pill badge-warning");

}

else if (dt.Rows[0]["account\_status"].ToString().Trim() == "deactive")

{

Label1.Attributes.Add("class", "badge badge-pill badge-danger");

}

else

{

Label1.Attributes.Add("class", "badge badge-pill badge-info");

}

}

catch (Exception ex)

{

Response.Write("<script>alert('" + ex.Message + "');</script>");

}

}

void getUserBookData()

{

try

{

SqlConnection con = new SqlConnection(strcon);

if (con.State == ConnectionState.Closed)

{

con.Open();

}

SqlCommand cmd = new SqlCommand("SELECT \* from book\_issue\_tbl where member\_id='" + Session["username"].ToString() + "';", con);

SqlDataAdapter da = new SqlDataAdapter(cmd);

DataTable dt = new DataTable();

da.Fill(dt);

GridView1.DataSource = dt;

GridView1.DataBind();

}

catch (Exception ex)

{

Response.Write("<script>alert('" + ex.Message + "');</script>");

}

}

protected void GridView1\_RowDataBound(object sender, GridViewRowEventArgs e)

{

try

{

if (e.Row.RowType == DataControlRowType.DataRow)

{

//Check your condition here

DateTime dt = Convert.ToDateTime(e.Row.Cells[5].Text);

DateTime today = DateTime.Today;

if (today > dt)

{

e.Row.BackColor = System.Drawing.Color.PaleVioletRed;

}

}

}

catch (Exception ex)

{

Response.Write("<script>alert('" + ex.Message + "');</script>");

}

}

}

}

#### Master page Design:-

1. using System;
2. using System.Collections.Generic;
3. using System.Linq;
4. using System.Web;
5. using System.Web.UI;
6. using System.Web.UI.WebControls;
7. namespace E\_library
8. {
9. public partial class Site1 : System.Web.UI.MasterPage
10. {
11. protected void Page\_Load(object sender, EventArgs e)
12. {
13. try
14. {
15. //This condition will check if anybody is log in or not
16. if (Session["role"] == null)
17. {
18. //this will show userlogin to the users
19. LinkButton1.Visible = true; //Userlogin link button
20. LinkButton2.Visible = true; //signup button link button
21. LinkButton3.Visible = false; //logout link button
22. LinkButton7.Visible = false; //hello user button link button
24. //this link will onl show to only admin except linkbutton 6 -->
25. LinkButton6.Visible = true; //admin login link button
26. LinkButton11.Visible = false; //author button link button
27. LinkButton12.Visible = false; //publisher link button
28. LinkButton8.Visible = false; //book inventory link button
29. LinkButton9.Visible = false; //book issuing link button
30. LinkButton10.Visible = false; //member management button link button
31. }
32. else if (Session["role"].Equals("user"))
33. {
34. //this will show userlogin to the users
35. LinkButton1.Visible = false; //Userlogin link button
36. LinkButton2.Visible = false; //signup button link button
37. LinkButton3.Visible = true; //logout link button
38. LinkButton7.Visible = true; //hello user button link button
39. LinkButton7.Text = "hello " + Session["username"].ToString();
40. //this condition will hide admin button which will onl be access by user
42. //this link will onl show to only admin
43. LinkButton6.Visible = true; //admin login link button
44. LinkButton11.Visible = false; //author button link button
45. LinkButton12.Visible = false; //publisher link button
46. LinkButton10.Visible = false; //member management button link button
47. }
48. else if (Session["role"].Equals("admin"))
49. {
50. //this will show userlogin to the users
51. LinkButton1.Visible = false; //Userlogin link button
52. LinkButton2.Visible = false; //signup button link button
53. LinkButton3.Visible = true; //logout link button
54. LinkButton7.Visible = true; //hello user button link button
55. LinkButton7.Text = "HELLO SUPERUSER! " + Session["username"].ToString();
57. //this link will onl show to only admin as this is admin login all will be true except 1 one
58. LinkButton6.Visible = false; //admin login link button
59. LinkButton11.Visible = true; //author button link button
60. LinkButton12.Visible = true; //publisher link button
61. LinkButton8.Visible = true; // book inventory link button
62. LinkButton9.Visible = true; // book issuing link button
63. LinkButton10.Visible = true; //member management button link button
64. }
65. }
66. catch (Exception ex)
67. {
68. Response.Write("<script>alert('" + ex.Message + "');</script>");
69. }
70. }
72. protected void LinkButton6\_Click(object sender, EventArgs e)
73. {
74. Response.Redirect("Adminlogin.aspx");
75. }
76. protected void LinkButton11\_Click(object sender, EventArgs e)
77. {
78. Response.Redirect("adminauthormanagement.aspx");
79. }
80. protected void LinkButton12\_Click(object sender, EventArgs e)
81. {
82. Response.Redirect("adminpublishermanagement.aspx");
83. }
84. protected void LinkButton8\_Click(object sender, EventArgs e)
85. {
86. Response.Redirect("adminbookinventory.aspx");
87. }
88. protected void LinkButton9\_Click(object sender, EventArgs e)
89. {
90. Response.Redirect("adminbookissuing.aspx");
91. }
92. protected void LinkButton10\_Click(object sender, EventArgs e)
93. {
94. Response.Redirect("adminmembermanagement.aspx");
95. }
96. protected void LinkButton3\_Click(object sender, EventArgs e)
97. {
98. Session["username"] = null;
99. Session["fullname"] = null;
100. Session["role"] = null;
101. Session["status"] = null;
102. //this will show userlogin to the users
103. LinkButton1.Visible = true; //Userlogin link button
104. LinkButton2.Visible = true; //signup button link button
105. LinkButton3.Visible = false; //logout link button
106. LinkButton7.Visible = false; //hello user button link button
107. //this link will onl show to only admin except linkbutton 6 -->
108. LinkButton6.Visible = true; //admin login link button
109. LinkButton11.Visible = false; //author button link button
110. LinkButton12.Visible = false; //publisher link button
111. LinkButton8.Visible = false; //book inventory link button
112. LinkButton9.Visible = false; //book issuing link button
113. LinkButton10.Visible = false; //member management button link button
114. }
115. }

#### }

#### MasterPage code:-

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

namespace E\_library

{

public partial class Site1 : System.Web.UI.MasterPage

{

protected void Page\_Load(object sender, EventArgs e)

{

try

{

//This condition will check if anybody is log in or not

if (Session["role"] == null)

{

//this will show userlogin to the users

LinkButton1.Visible = true; //Userlogin link button

LinkButton2.Visible = true; //signup button link button

LinkButton3.Visible = false; //logout link button

LinkButton7.Visible = false; //hello user button link button

//this link will onl show to only admin except linkbutton 6 -->

LinkButton6.Visible = true; //admin login link button

LinkButton11.Visible = false; //author button link button

LinkButton12.Visible = false; //publisher link button

LinkButton8.Visible = false; //book inventory link button

LinkButton9.Visible = false; //book issuing link button

LinkButton10.Visible = false; //member management button link button

}

else if (Session["role"].Equals("user"))

{

//this will show userlogin to the users

LinkButton1.Visible = false; //Userlogin link button

LinkButton2.Visible = false; //signup button link button

LinkButton3.Visible = true; //logout link button

LinkButton7.Visible = true; //hello user button link button

LinkButton7.Text = "hello " + Session["username"].ToString();

//this condition will hide admin button which will onl be access by user

//this link will onl show to only admin

LinkButton6.Visible = true; //admin login link button

LinkButton11.Visible = false; //author button link button

LinkButton12.Visible = false; //publisher link button

LinkButton10.Visible = false; //member management button link button

}

else if (Session["role"].Equals("admin"))

{

//this will show userlogin to the users

LinkButton1.Visible = false; //Userlogin link button

LinkButton2.Visible = false; //signup button link button

LinkButton3.Visible = true; //logout link button

LinkButton7.Visible = true; //hello user button link button

LinkButton7.Text = "HELLO SUPERUSER! " + Session["username"].ToString();

//this link will onl show to only admin as this is admin login all will be true except 1 one

LinkButton6.Visible = false; //admin login link button

LinkButton11.Visible = true; //author button link button

LinkButton12.Visible = true; //publisher link button

LinkButton8.Visible = true; // book inventory link button

LinkButton9.Visible = true; // book issuing link button

LinkButton10.Visible = true; //member management button link button

}

}

catch (Exception ex)

{

Response.Write("<script>alert('" + ex.Message + "');</script>");

}

}

protected void LinkButton6\_Click(object sender, EventArgs e)

{

Response.Redirect("Adminlogin.aspx");

}

protected void LinkButton11\_Click(object sender, EventArgs e)

{

Response.Redirect("adminauthormanagement.aspx");

}

protected void LinkButton12\_Click(object sender, EventArgs e)

{

Response.Redirect("adminpublishermanagement.aspx");

}

protected void LinkButton8\_Click(object sender, EventArgs e)

{

Response.Redirect("adminbookinventory.aspx");

}

protected void LinkButton9\_Click(object sender, EventArgs e)

{

Response.Redirect("adminbookissuing.aspx");

}

protected void LinkButton10\_Click(object sender, EventArgs e)

{

Response.Redirect("adminmembermanagement.aspx");

}

protected void LinkButton3\_Click(object sender, EventArgs e)

{

Session["username"] = null;

Session["fullname"] = null;

Session["role"] = null;

Session["status"] = null;

//this will show userlogin to the users

LinkButton1.Visible = true; //Userlogin link button

LinkButton2.Visible = true; //signup button link button

LinkButton3.Visible = false; //logout link button

LinkButton7.Visible = false; //hello user button link button

//this link will onl show to only admin except linkbutton 6 -->

LinkButton6.Visible = true; //admin login link button

LinkButton11.Visible = false; //author button link button

LinkButton12.Visible = false; //publisher link button

LinkButton8.Visible = false; //book inventory link button

LinkButton9.Visible = false; //book issuing link button

LinkButton10.Visible = false; //member management button link button

}

}

#### }

#### Code For Author Management design:-

<%@ Page Title="" Language="C#" MasterPageFile="~/Site1.Master" AutoEventWireup="true" CodeBehind="adminauthormanagement.aspx.cs" Inherits="E\_library.adminauthormanagement" %>

<asp:Content ID="Content1" ContentPlaceHolderID="head" runat="server">

<link href="Content/adminpuublish.css" rel="stylesheet" />

<script type="text/javascript">

$(document).ready(function () {

$(".table").prepend($("<thead></thead>").append($(this).find("tr:first"))).dataTable();

});

</script>>

</asp:Content>

<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1" runat="server">

<div class="container">

<div class="row">

<div class="col-md-5">

<div class="card">

<div class="card-body">

<div class="row">

<div class="col">

<center>

<h4>Author Details</h4>

</center>

</div>

</div>

<div class="row">

<div class="col">

<center>

<img src="Images/writer.png" width="100px" />

</center>

</div>

</div>

<div class="row">

<div class="col">

<hr>

</div>

</div>

<div class="row">

<div class="col-md-4">

<label>Author ID</label>

<div class="form-group">

<div class="input-group">

<asp:TextBox CssClass="form-control" ID="TextBox1" runat="server" placeholder="ID"></asp:TextBox>

<asp:Button class="btn btn-primary" ID="Button1" runat="server" Text="Go" OnClick="Button1\_Click" />

</div>

</div>

</div>

<div class="col-md-8">

<label> Author Name </label>

<div class="form-group">

<asp:TextBox CssClass="form-control" ID="TextBox2" runat="server" placeholder="Author Name"></asp:TextBox>

</div>

</div>

</div>

<br />

<div class="row">

<div class="col-4">

<asp:Button ID="Button2" class="btn btn-lg btn-block btn-success" runat="server" Text="Add" OnClick="Button2\_Click" />

</div>

<div class="col-4">

<asp:Button ID="Button3" class="btn btn-lg btn-block btn-warning" runat="server" Text="Update" OnClick="Button3\_Click" />

</div>

<div class="col-4">

<asp:Button ID="Button4" class="btn btn-lg btn-block btn-danger" runat="server" Text="Delete" OnClick="Button4\_Click" />

</div>

</div>

</div>

</div>

<a href="homePage.aspx"> << Back to Home</a><br>

<br>

</div>

<div class="col-md-7">

<div class="card">

<div class="card-body">

<div class="row">

<div class="col">

<center>

<h4>Author List</h4>

</center>

</div>

</div>

<div class="row">

<div class="col">

<hr>

</div>

</div>

<div class="row">

<asp:SqlDataSource ID="SqlDataSource1" runat="server" ConnectionString="<%$ ConnectionStrings:elibraryDBConnectionString %>" SelectCommand="SELECT \* FROM [author\_master\_tbl]"></asp:SqlDataSource>

<div class="col">

<asp:GridView class="table table-striped table-bordered" ID="GridView1" runat="server" AutoGenerateColumns="False" DataKeyNames="author\_id" DataSourceID="SqlDataSource1" >

<Columns>

<asp:BoundField DataField="author\_id" HeaderText="author\_id" ReadOnly="True" SortExpression="author\_id" />

<asp:BoundField DataField="author\_name" HeaderText="author\_name" SortExpression="author\_name" />

</Columns>

</asp:GridView>

</div>

</div>

</div>

</div>

</div>

</div>

</div>

</asp:Content>

#### Code For Author management System:-

using System;

using System.Collections.Generic;

using System.Configuration;

using System.Data;

using System.Data.SqlClient;

using System.Linq;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

namespace E\_library

{

public partial class adminauthormanagement : System.Web.UI.Page

{

String strcon = ConfigurationManager.ConnectionStrings["con"].ConnectionString;

protected void Page\_Load(object sender, EventArgs e)

{

GridView1.DataBind();

}

//add Button click //issue book

protected void Button2\_Click(object sender, EventArgs e)

{

if (checkIfAuthorExists())

{

Response.Write("<script>alert('author with this ID already exists... you cannot add another author with same Author ID ');</script>");

}

else

{

AddNewAuthor();

}

}

//update Button Click

protected void Button3\_Click(object sender, EventArgs e)

{

if (checkIfAuthorExists())

{

updateAuthor();

}

else

{

Response.Write("<script>alert('Author Does Not Exist... ');</script>");

}

}

//Delete Button Click

protected void Button4\_Click(object sender, EventArgs e)

{

if (checkIfAuthorExists())

{

DeleteAuthor();

}

else

{

Response.Write("<script>alert('Author Does Not Exist... ');</script>");

}

}

//Go button click

protected void Button1\_Click(object sender, EventArgs e)

{

getAuthorById();

}

//User DEfined Function

//go buttom

void getAuthorById()

{

try

{

SqlConnection con = new SqlConnection(strcon);

if (con.State == ConnectionState.Closed)

{

con.Open();

}

SqlCommand cmd = new SqlCommand("SELECT \* from author\_master\_tbl where author\_id='" + TextBox1.Text.Trim() + "'", con);

SqlDataAdapter da = new SqlDataAdapter(cmd);

DataTable dt = new DataTable();

da.Fill(dt);

if (dt.Rows.Count >= 1)

{

TextBox2.Text = dt.Rows[0][1].ToString();

}

else

{

Response.Write("<script>alert('INVALID AUTHOR ID')</script>");

}

}

catch (Exception ex)

{

Response.Write("<script>alert('" + ex.Message + "');</script>");

}

}

//to delete author

void DeleteAuthor()

{

try

{

SqlConnection con = new SqlConnection(strcon);

if (con.State == ConnectionState.Closed)

{

con.Open();

}

SqlCommand cmd = new SqlCommand("DELETE from author\_master\_tbl WHERE author\_id='" + TextBox1.Text.Trim() + "'", con);

cmd.ExecuteNonQuery();

con.Close();

Response.Write("<script>alert('Author Deleted Successfully ');</script>");

clearForm();

GridView1.DataBind();

}

catch (Exception ex)

{

Response.Write("<script>alert('" + ex.Message + "');</script>");

}

}

//for Updating Author detail/Name

void updateAuthor()

{

try

{

SqlConnection con = new SqlConnection(strcon);

if (con.State == ConnectionState.Closed)

{

con.Open();

}

SqlCommand cmd = new SqlCommand("UPDATE author\_master\_tbl SET author\_name = @author\_name WHERE author\_id='"+TextBox1.Text.Trim()+"'", con);

//basically values are parameter which can be anthing but use same

cmd.Parameters.AddWithValue("@author\_name", TextBox2.Text.Trim());

cmd.ExecuteNonQuery();

con.Close();

Response.Write("<script>alert('Author Updated Successfully ');</script>");

clearForm();

GridView1.DataBind();

}

catch (Exception ex)

{

Response.Write("<script>alert('" + ex.Message + "');</script>");

}

}

//for adding new author

void AddNewAuthor()

{

try

{

SqlConnection con = new SqlConnection(strcon);

if (con.State == ConnectionState.Closed)

{

con.Open();

}

SqlCommand cmd = new SqlCommand("INSERT INTO author\_master\_tbl(author\_id,author\_name) values(@author\_id,@author\_name) ", con);

cmd.Parameters.AddWithValue("@author\_id", TextBox1.Text.Trim()); //basically values are parameter which can be anthing but use same

cmd.Parameters.AddWithValue("@author\_name", TextBox2.Text.Trim());

cmd.ExecuteNonQuery();

con.Close();

Response.Write("<script>alert('Author added Successfully ');</script>");

clearForm();

GridView1.DataBind();

}

catch (Exception ex)

{

Response.Write("<script>alert('" + ex.Message + "');</script>");

}

}

//for chechking if Author is present or not

bool checkIfAuthorExists()

{

try

{

SqlConnection con = new SqlConnection(strcon);

if (con.State == ConnectionState.Closed)

{

con.Open();

}

SqlCommand cmd = new SqlCommand("SELECT \* from author\_master\_tbl where author\_id='" + TextBox1.Text.Trim() + "'", con);

SqlDataAdapter da = new SqlDataAdapter(cmd);

DataTable dt = new DataTable();

da.Fill(dt);

if (dt.Rows.Count >= 1)

{

return true;

}

else

{

return false;

}

}

catch (Exception ex)

{

Response.Write("<script>alert('" + ex.Message + "');</script>");

return false;

}

}

void clearForm()

{

TextBox1.Text = "";

TextBox2.Text = "";

}

}

#### }

#### Book Inventory Design :-

#### <%@ Page Title="" Language="C#" MasterPageFile="~/Site1.Master" AutoEventWireup="true" CodeBehind="adminbookinventory.aspx.cs" Inherits="E\_library.adminbookinventory" %>

#### <asp:Content ID="Content1" ContentPlaceHolderID="head" runat="server">

#### <link href="Content/adminbookinventory/StyleSheet1.css" rel="stylesheet" />

#### <script type="text/javascript" >

#### $(document).ready(function () {

#### $(".table").prepend($("<thead></thead>").append($(this).find("tr:first"))).dataTable();

#### });

#### function readURL(input) {

#### if (input.files && input.files[0]) {

#### var reader = new FileReader();

#### reader.onload = function (e) {

#### $('#imgview').attr('src', e.target.result);

#### };

#### reader.readAsDataURL(input.files[0]);

#### }

#### }

#### </script>

#### </asp:Content>

#### <asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1" runat="server">

#### <div class="container-fluid">

#### <div class="row">

#### <div class="col-md-5">

#### <div class="card">

#### <div class="card-body">

#### <div class="row">

#### <div class="col">

#### <center>

#### <h4>Book Details</h4>

#### </center>

#### </div>

#### </div>

#### <div class="row">

#### <div class="col">

#### <center>

#### <img src="Images/books.png" width="100px" />

#### </center>

#### </div>

#### </div>

#### <div class="row">

#### <div class="col">

#### <hr>

#### </div>

#### </div>

#### <div class="row">

#### <div class="col">

#### <asp:FileUpload onchange="readURL(this);" class="form-control" ID="FileUpload1" runat="server" />

#### </div>

#### </div>

#### <div class="row">

#### <div class="col-md-4">

#### <label>Book ID</label>

#### <div class="form-group">

#### <div class="input-group">

#### <asp:TextBox CssClass="form-control" ID="TextBox1" runat="server" placeholder="Book ID"></asp:TextBox>

#### <asp:Button class="btn btn-primary" ID="button4" runat="server" Text="GO" OnClick="button4\_Click" />

#### </div>

#### </div>

#### </div>

#### <div class="col-md-8">

#### <label>Book Name</label>

#### <div class="form-group">

#### <asp:TextBox CssClass="form-control" ID="TextBox2" runat="server" placeholder="Book Name"></asp:TextBox>

#### </div>

#### </div>

#### </div>

#### <div class="row">

#### <div class="col-md-4">

#### <label>Language</label>

#### <div class="form-group">

#### <asp:DropDownList class="form-control" ID="DropDownList1" runat="server">

#### <asp:ListItem Text="English" Value="English" />

#### <asp:ListItem Text="Hindi" Value="Hindi" />

#### <asp:ListItem Text="Marathi" Value="Marathi" />

#### <asp:ListItem Text="French" Value="French" />

#### <asp:ListItem Text="German" Value="German" />

#### <asp:ListItem Text="Urdu" Value="Urdu" />

#### </asp:DropDownList>

#### </div>

#### <label>Publisher Name</label>

#### <div class="form-group">

#### <asp:DropDownList class="form-control" ID="DropDownList2" runat="server">

#### <asp:ListItem Text="Publisher 1" Value="Publisher 1" />

#### <asp:ListItem Text="Publisher 2" Value="Publisher 2" />

#### </asp:DropDownList>

#### </div>

#### </div>

#### <div class="col-md-4">

#### <label>Author Name</label>

#### <div class="form-group">

#### <asp:DropDownList class="form-control" ID="DropDownList3" runat="server">

#### <asp:ListItem Text="A1" Value="a1" />

#### <asp:ListItem Text="a2" Value="a2" />

#### </asp:DropDownList>

#### </div>

#### <label>Publish Date</label>

#### <div class="form-group">

#### <asp:TextBox CssClass="form-control" ID="TextBox3" runat="server" placeholder="Date" TextMode="Date"></asp:TextBox>

#### </div>

#### </div>

#### <div class="col-md-4">

#### <label>Genre</label>

#### <div class="form-group">

#### <asp:ListBox CssClass="form-control" ID="ListBox1" runat="server" SelectionMode="Multiple" Rows="5">

#### <asp:ListItem Text="Action" Value="Action" />

#### <asp:ListItem Text="Adventure" Value="Adventure" />

#### <asp:ListItem Text="Comic Book" Value="Comic Book" />

#### <asp:ListItem Text="Self Help" Value="Self Help" />

#### <asp:ListItem Text="Motivation" Value="Motivation" />

#### <asp:ListItem Text="Healthy Living" Value="Healthy Living" />

#### <asp:ListItem Text="Wellness" Value="Wellness" />

#### <asp:ListItem Text="Crime" Value="Crime" />

#### <asp:ListItem Text="Drama" Value="Drama" />

#### <asp:ListItem Text="Fantasy" Value="Fantasy" />

#### <asp:ListItem Text="Horror" Value="Horror" />

#### <asp:ListItem Text="Poetry" Value="Poetry" />

#### <asp:ListItem Text="Personal Development" Value="Personal Development" />

#### <asp:ListItem Text="Romance" Value="Romance" />

#### <asp:ListItem Text="Science Fiction" Value="Science Fiction" />

#### <asp:ListItem Text="Suspense" Value="Suspense" />

#### <asp:ListItem Text="Thriller" Value="Thriller" />

#### <asp:ListItem Text="Art" Value="Art" />

#### <asp:ListItem Text="Autobiography" Value="Autobiography" />

#### <asp:ListItem Text="Encyclopedia" Value="Encyclopedia" />

#### <asp:ListItem Text="Health" Value="Health" />

#### <asp:ListItem Text="History" Value="History" />

#### <asp:ListItem Text="Math" Value="Math" />

#### <asp:ListItem Text="Textbook" Value="Textbook" />

#### <asp:ListItem Text="Science" Value="Science" />

#### <asp:ListItem Text="Travel" Value="Travel" />

#### </asp:ListBox>

#### </div>

#### </div>

#### </div>

#### <div class="row">

#### <div class="col-md-4">

#### <label>Edition</label>

#### <div class="form-group">

#### <asp:TextBox CssClass="form-control" ID="TextBox9" runat="server" placeholder="Edition"></asp:TextBox>

#### </div>

#### </div>

#### <div class="col-md-4">

#### <label>Book Cost(per unit)</label>

#### <div class="form-group">

#### <asp:TextBox CssClass="form-control" ID="TextBox10" runat="server" placeholder="Book Cost(per unit)" TextMode="Number"></asp:TextBox>

#### </div>

#### </div>

#### <div class="col-md-4">

#### <label>Pages</label>

#### <div class="form-group">

#### <asp:TextBox CssClass="form-control" ID="TextBox11" runat="server" placeholder="Pages" TextMode="Number"></asp:TextBox>

#### </div>

#### </div>

#### </div>

#### <div class="row">

#### <div class="col-md-4">

#### <label>Actual Stock</label>

#### <div class="form-group">

#### <asp:TextBox CssClass="form-control" ID="TextBox4" runat="server" placeholder="Actual Stock" TextMode="Number"></asp:TextBox>

#### </div>

#### </div>

#### <div class="col-md-4">

#### <label>Current Stock</label>

#### <div class="form-group">

#### <asp:TextBox CssClass="form-control" ID="TextBox5" runat="server" placeholder="Book Cost(per unit)" TextMode="Number" ReadOnly="True"></asp:TextBox>

#### </div>

#### </div>

#### <div class="col-md-4">

#### <label>Issued Books</label>

#### <div class="form-group">

#### <asp:TextBox CssClass="form-control" ID="TextBox7" runat="server" placeholder="Pages" TextMode="Number" ReadOnly="True"></asp:TextBox>

#### </div>

#### </div>

#### </div>

#### <div class="row">

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

#### <label>Book Description</label>

#### <div class="form-group">

#### <asp:TextBox CssClass="form-control" ID="TextBox6" runat="server" placeholder="Book Description" TextMode="MultiLine" Rows="2"></asp:TextBox>

#### </div>

#### </div>

#### </div>

#### <div class="row">

#### <div class="col-4">

#### <asp:Button ID="Button1" class="btn btn-lg btn-block btn-success" runat="server" Text="Add" OnClick="Button1\_Click" />

#### </div>

#### <div class="col-4">

#### <asp:Button ID="Button3" class="btn btn-lg btn-block btn-warning" runat="server" Text="Update" OnClick="Button3\_Click" />

#### </div>

#### <div class="col-4">

#### <asp:Button ID="Button2" class="btn btn-lg btn-block btn-danger" runat="server" Text="Delete" OnClick="Button2\_Click" />

#### </div>

#### </div>

#### </div>

#### </div>

#### <a href="homepage.aspx"><< Back to Home</a><br>

#### <br>

#### </div>

#### <div class="col-md-7">

#### <div class="card">

#### <div class="card-body">

#### <div class="row">

#### <div class="col">

#### <center>

#### <h4>Book Inventory List</h4>

#### </center>

#### </div>

#### </div>

#### <div class="row">

#### <div class="col">

#### <hr>

#### </div>

#### </div>

#### <div class="row">

#### <asp:SqlDataSource ID="SqlDataSource1" runat="server" ConnectionString="<%$ ConnectionStrings:elibraryDBConnectionString %>" SelectCommand="SELECT \* FROM [book\_master\_tbl]"></asp:SqlDataSource>

#### <div class="col">

#### <asp:GridView class="table table-striped table-bordered" ID="GridView1" runat="server" AutoGenerateColumns="False" DataKeyNames="book\_id" DataSourceID="SqlDataSource1">

#### <Columns>

#### <asp:BoundField DataField="book\_id" HeaderText="ID" ReadOnly="True" SortExpression="book\_id" >

#### 

#### <ItemStyle Font-Bold="True" />

#### </asp:BoundField>

#### 

#### <asp:TemplateField>

#### <ItemTemplate>

#### <div class="container-fluid">

#### <div class="row">

#### <div class="col-lg-10">

#### <div class="row">

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

#### <asp:Label ID="label1" runat="server" Text='<%# Eval("[book name]") %>' Font-Bold="True" Font-Size="X-Large"></asp:Label>

#### </div>

#### </div>

#### <div class="row">

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

#### Author -<asp:Label ID="Label2" runat="server" Font-Bold="True" Text='<%# Eval("author\_name") %>'></asp:Label>

#### &nbsp;| Genre -<asp:Label ID="Label3" runat="server" Font-Bold="True" Text='<%# Eval("genre") %>'></asp:Label>

#### &nbsp;| Language -

#### <asp:Label ID="Label4" runat="server" Font-Bold="True" Text='<%# Eval("language") %>'></asp:Label>

#### 

#### </div>

#### </div>

#### <div class="row">

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

#### Publisher -

#### <asp:Label ID="Label5" runat="server" Font-Bold="True" Text='<%# Eval("publisher\_name") %>'></asp:Label>

#### &nbsp;| Publish Date-<asp:Label ID="Label6" runat="server" Font-Bold="True" Text='<%# Eval("publish\_date") %>'></asp:Label>

#### &nbsp;| Pages -

#### <asp:Label ID="Label7" runat="server" Font-Bold="True" Text='<%# Eval("no\_of\_pages") %>'></asp:Label>

#### &nbsp;| Edition -

#### <asp:Label ID="Label8" runat="server" Font-Bold="True" Text='<%# Eval("edition") %>'></asp:Label>

#### </div>

#### </div>

#### <div class="row">

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

#### cost -

#### <asp:Label ID="Label9" runat="server" Font-Bold="True" Text='<%# Eval("book\_cost") %>'></asp:Label>

#### &nbsp;| Actual Stock -

#### <asp:Label ID="Label10" runat="server" Font-Bold="True" Text='<%# Eval("actual\_stock") %>'></asp:Label>

#### &nbsp;| Available stock -

#### <asp:Label ID="Label11" runat="server" Font-Bold="True" Text='<%# Eval("current\_stock") %>'></asp:Label>

#### </div>

#### </div>

#### <div class="row">

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

#### Description -

#### <asp:Label ID="Label12" runat="server" Font-Bold="True" Font-Italic="True" Font-Size="Smaller" Text='<%# Eval("book\_description") %>'></asp:Label>

#### </div>

#### </div>

#### </div>

#### <div class="col-lg-2">

#### <asp:Image ID="Image1" class="img-fluid " runat="server" ImageUrl='<%# Eval("book\_img\_link") %>' />

#### </div>

#### </div>

#### </div>

#### 

#### </div>

#### </ItemTemplate>

#### </asp:TemplateField>

#### 

#### </Columns>

#### </asp:GridView>

#### </div>

#### </div>

#### </div>

#### </div>

#### </div>

#### </div>

#### </div>

#### </asp:Content>

#### Code For Book Inventory:-

using System;

using System.Collections.Generic;

using System.Linq;

using System.Configuration;

using System.Data;

using System.Data.SqlClient;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

using System.IO;

namespace E\_library

{

public partial class adminbookinventory : System.Web.UI.Page

{

string strcon = ConfigurationManager.ConnectionStrings["con"].ConnectionString;

static string global\_filepath;

static int global\_actual\_stock, global\_current\_stock, global\_issued\_books;

protected void Page\_Load(object sender, EventArgs e)

{

if (!IsPostBack)

{

fillAuthorPublishervalues();

}

GridView1.DataBind();

}

//go button -->

protected void button4\_Click(object sender, EventArgs e)

{

getBookById();

}

//add button -->

protected void Button1\_Click(object sender, EventArgs e)

{

if (checkIfBookExists())

{

Response.Write("<script>alert('Book Already Exists ,Try some other Book ID');</script>"); //

}

else

{

addNewBook();

}

}

//update button -->

protected void Button3\_Click(object sender, EventArgs e)

{

updateBookByID();

}

//Delete button -->

protected void Button2\_Click(object sender, EventArgs e)

{

deleteBookByID();

}

//User defined function

void getBookById()

{

try

{

SqlConnection con = new SqlConnection(strcon);

if (con.State == ConnectionState.Closed)

{

con.Open();

}

SqlCommand cmd = new SqlCommand("SELECT \* from book\_master\_tbl WHERE book\_id = '"+TextBox1.Text.Trim()+"' ;", con);

SqlDataAdapter da = new SqlDataAdapter(cmd);

DataTable dt = new DataTable();

da.Fill(dt);

if(dt.Rows.Count >= 1)

{

TextBox2.Text = dt.Rows[0]["book\_name"].ToString();

TextBox3.Text = dt.Rows[0]["publish\_date"].ToString();

TextBox9.Text = dt.Rows[0]["edition"].ToString();

TextBox10.Text = dt.Rows[0]["book\_cost"].ToString();

TextBox11.Text = dt.Rows[0]["no\_of\_pages"].ToString();

TextBox4.Text = dt.Rows[0]["actual\_stock"].ToString();

TextBox5.Text = dt.Rows[0]["current\_stock"].ToString();

TextBox6.Text = dt.Rows[0]["book\_description"].ToString();

TextBox7.Text = "" + (Convert.ToInt32(dt.Rows[0]["actual\_stock"].ToString()) - Convert.ToInt32(dt.Rows[0]["current\_stock"].ToString()));

DropDownList1.SelectedValue = dt.Rows[0]["publisher\_name"].ToString().Trim();

DropDownList2.SelectedValue = dt.Rows[0]["language"].ToString().Trim();

DropDownList3.SelectedValue = dt.Rows[0]["author\_name"].ToString().Trim();

ListBox1.ClearSelection();

string[] genre = dt.Rows[0]["genre"].ToString().Trim().Split(',');

for(int i = 0; i < genre.Length; i++)

{

for(int j = 0; j < ListBox1.Items.Count; j++)

{

if (ListBox1.Items[j].ToString() == genre[i])

{

ListBox1.Items[j].Selected = true;

}

}

}

}

else

{

Response.Write("<script>alert(''Book added successfully);</script>");

}

}

catch (Exception ex)

{

}

}

//delete book ID

void deleteBookByID()

{

if (checkIfBookExists())

{

try

{

SqlConnection con = new SqlConnection(strcon);

if (con.State == ConnectionState.Closed)

{

con.Open();

}

SqlCommand cmd = new SqlCommand("DELETE from book\_master\_tbl WHERE book\_id='" + TextBox1.Text.Trim() + "'", con);

cmd.ExecuteNonQuery();

con.Close();

Response.Write("<script>alert('Book Deleted Successfully');</script>");

GridView1.DataBind();

}

catch (Exception ex)

{

Response.Write("<script>alert('" + ex.Message + "');</script>");

}

}

else

{

Response.Write("<script>alert('Invalid Member ID');</script>");

}

}

//Update Book ID

void updateBookByID()

{

if (checkIfBookExists())

{

try

{

int actual\_stock = Convert.ToInt32(TextBox4.Text.Trim());

int current\_stock = Convert.ToInt32(TextBox5.Text.Trim());

if (global\_actual\_stock == actual\_stock)

{

}

else

{

if (actual\_stock < global\_issued\_books)

{

Response.Write("<script>alert('Actual Stock value cannot be less than the Issued books');</script>");

return;

}

else

{

current\_stock = actual\_stock - global\_issued\_books;

TextBox5.Text = "" + current\_stock;

}

}

string genres = "";

foreach (int i in ListBox1.GetSelectedIndices())

{

genres = genres + ListBox1.Items[i] + ",";

}

genres = genres.Remove(genres.Length - 1);

string filepath = "~/book\_inventory/books1";

string filename = Path.GetFileName(FileUpload1.PostedFile.FileName);

if (filename == "" || filename == null)

{

filepath = global\_filepath;

}

else

{

FileUpload1.SaveAs(Server.MapPath("book\_inventory/" + filename));

filepath = "~/book\_inventory/" + filename;

}

SqlConnection con = new SqlConnection(strcon);

if (con.State == ConnectionState.Closed)

{

con.Open();

}

SqlCommand cmd = new SqlCommand("UPDATE book\_master\_tbl set book\_name=@book\_name, genre=@genre, author\_name=@author\_name, publisher\_name=@publisher\_name, publish\_date=@publish\_date, language=@language, edition=@edition, book\_cost=@book\_cost, no\_of\_pages=@no\_of\_pages, book\_description=@book\_description, actual\_stock=@actual\_stock, current\_stock=@current\_stock, book\_img\_link=@book\_img\_link where book\_id='" + TextBox1.Text.Trim() + "'", con);

cmd.Parameters.AddWithValue("@book\_name", TextBox2.Text.Trim());

cmd.Parameters.AddWithValue("@genre", genres);

cmd.Parameters.AddWithValue("@author\_name", DropDownList3.SelectedItem.Value);

cmd.Parameters.AddWithValue("@publisher\_name", DropDownList2.SelectedItem.Value);

cmd.Parameters.AddWithValue("@publish\_date", TextBox3.Text.Trim());

cmd.Parameters.AddWithValue("@language", DropDownList1.SelectedItem.Value);

cmd.Parameters.AddWithValue("@edition", TextBox9.Text.Trim());

cmd.Parameters.AddWithValue("@book\_cost", TextBox10.Text.Trim());

cmd.Parameters.AddWithValue("@no\_of\_pages", TextBox11.Text.Trim());

cmd.Parameters.AddWithValue("@book\_description", TextBox6.Text.Trim());

cmd.Parameters.AddWithValue("@actual\_stock", actual\_stock.ToString());

cmd.Parameters.AddWithValue("@current\_stock", current\_stock.ToString());

cmd.Parameters.AddWithValue("@book\_img\_link", filepath);

cmd.ExecuteNonQuery();

con.Close();

GridView1.DataBind();

Response.Write("<script>alert('Book Updated Successfully');</script>");

}

catch (Exception ex)

{

Response.Write("<script>alert('" + ex.Message + "');</script>");

}

}

else

{

Response.Write("<script>alert('Invalid Book ID');</script>");

}

}

//get books by ID

void getBookByID()

{

try

{

SqlConnection con = new SqlConnection(strcon);

if (con.State == ConnectionState.Closed)

{

con.Open();

}

SqlCommand cmd = new SqlCommand("SELECT \* from book\_master\_tbl WHERE book\_id='" + TextBox1.Text.Trim() + "';", con);

SqlDataAdapter da = new SqlDataAdapter(cmd);

DataTable dt = new DataTable();

da.Fill(dt);

if (dt.Rows.Count >= 1)

{

TextBox2.Text = dt.Rows[0]["book\_name"].ToString();

TextBox3.Text = dt.Rows[0]["publish\_date"].ToString();

TextBox9.Text = dt.Rows[0]["edition"].ToString();

TextBox10.Text = dt.Rows[0]["book\_cost"].ToString().Trim();

TextBox11.Text = dt.Rows[0]["no\_of\_pages"].ToString().Trim();

TextBox4.Text = dt.Rows[0]["actual\_stock"].ToString().Trim();

TextBox5.Text = dt.Rows[0]["current\_stock"].ToString().Trim();

TextBox6.Text = dt.Rows[0]["book\_description"].ToString();

TextBox7.Text = "" + (Convert.ToInt32(dt.Rows[0]["actual\_stock"].ToString()) - Convert.ToInt32(dt.Rows[0]["current\_stock"].ToString()));

DropDownList1.SelectedValue = dt.Rows[0]["language"].ToString().Trim();

DropDownList2.SelectedValue = dt.Rows[0]["publisher\_name"].ToString().Trim();

DropDownList3.SelectedValue = dt.Rows[0]["author\_name"].ToString().Trim();

ListBox1.ClearSelection();

string[] genre = dt.Rows[0]["genre"].ToString().Trim().Split(',');

for (int i = 0; i < genre.Length; i++)

{

for (int j = 0; j < ListBox1.Items.Count; j++)

{

if (ListBox1.Items[j].ToString() == genre[i])

{

ListBox1.Items[j].Selected = true;

}

}

}

global\_actual\_stock = Convert.ToInt32(dt.Rows[0]["actual\_stock"].ToString().Trim());

global\_current\_stock = Convert.ToInt32(dt.Rows[0]["current\_stock"].ToString().Trim());

global\_issued\_books = global\_actual\_stock - global\_current\_stock;

global\_filepath = dt.Rows[0]["book\_img\_link"].ToString();

}

else

{

Response.Write("<script>alert('Invalid Book ID');</script>");

}

}

catch (Exception ex)

{

}

}

//Author Publisher View

void fillAuthorPublishervalues()

{

try

{

SqlConnection con = new SqlConnection(strcon);

if (con.State == ConnectionState.Closed)

{

con.Open();

}

SqlCommand cmd = new SqlCommand("SELECT author\_name from author\_master\_tbl;", con);

SqlDataAdapter da = new SqlDataAdapter(cmd);

DataTable dt = new DataTable();

da.Fill(dt);

DropDownList3.DataSource = dt;

DropDownList3.DataValueField = "author\_name";

DropDownList3.DataBind();

cmd = new SqlCommand("SELECT publisher\_name from publisher\_master\_tbl;", con);

da = new SqlDataAdapter(cmd);

dt = new DataTable();

da.Fill(dt);

DropDownList2.DataSource = dt;

DropDownList2.DataValueField = "publisher\_name";

DropDownList2.DataBind();

}

catch (Exception ex)

{

}

}

bool checkIfBookExists()

{

try

{

SqlConnection con = new SqlConnection(strcon);

if (con.State == ConnectionState.Closed)

{

con.Open();

}

SqlCommand cmd = new SqlCommand("SELECT \* from book\_master\_tbl where book\_id='" + TextBox1.Text.Trim() + "'OR book\_name'" + TextBox2.Text.Trim() + "'", con);

SqlDataAdapter da = new SqlDataAdapter(cmd);

DataTable dt = new DataTable();

da.Fill(dt);

if (dt.Rows.Count >= 1)

{

return true;

}

else

{

return false;

}

}

catch (Exception ex)

{

Response.Write("<script>alert('" + ex.Message + "');</script>");

return false;

}

}

//add new books

void addNewBook()

{

try

{

//this is for multiple genre

string genres = "";

foreach (int i in ListBox1.GetSelectedIndices())

{

genres = genres + ListBox1.Items[i] + ",";

}

genres = genres.Remove(genres.Length - 1);

//file link

string filepath = "~/book\_inventory/books1.png";

string filename = Path.GetFileName(FileUpload1.PostedFile.FileName);

FileUpload1.SaveAs(Server.MapPath("book\_inventory/" + filename));

//all the details about book

SqlConnection con = new SqlConnection(strcon);

if (con.State == ConnectionState.Closed)

{

con.Open();

}

SqlCommand cmd = new SqlCommand("INSERT INTO book\_master\_tbl(book\_id,book\_name,genre,author\_name,publisher\_name,publisher\_date,language,edition,book\_cost,no\_of\_pages,book\_description,actual\_stock,current\_stock,book\_img\_link)values(@book\_id,@book\_name,@genre,@author\_name,@publisher\_name,@publisher\_date,@language,@edition,@book\_cost,@no\_of\_pages,@book\_description,@actual\_stock,@current\_stock,@book\_img\_link)", con);

cmd.Parameters.AddWithValue("@book\_id", TextBox1.Text.Trim());

cmd.Parameters.AddWithValue("@book\_name", TextBox2.Text.Trim());

cmd.Parameters.AddWithValue("@genre", genres);

cmd.Parameters.AddWithValue("@author\_name", DropDownList3.SelectedItem.Value);

cmd.Parameters.AddWithValue("@publisher\_name", DropDownList2.SelectedItem.Value);

cmd.Parameters.AddWithValue("@publish\_date", TextBox3.Text.Trim());

cmd.Parameters.AddWithValue("@language", DropDownList1.SelectedItem.Value);

cmd.Parameters.AddWithValue("@edition", TextBox9.Text.Trim());

cmd.Parameters.AddWithValue("@book\_cost", TextBox10.Text.Trim());

cmd.Parameters.AddWithValue("@no\_of\_pages", TextBox11.Text.Trim());

cmd.Parameters.AddWithValue("@book\_description", TextBox6.Text.Trim());

cmd.Parameters.AddWithValue("@actual\_stock", TextBox4.Text.Trim());

cmd.Parameters.AddWithValue("@current\_stock", TextBox4.Text.Trim());

cmd.Parameters.AddWithValue("@book\_img\_link", filepath);

cmd.ExecuteNonQuery();

con.Close();

Response.Write("<script>alert('Book added successfully.');</script>");

GridView1.DataBind();

}

catch (Exception ex)

{

Response.Write("<script> alert('" + ex.Message + "');</script>");

}

}

}

}

#### Design For Book Issueing:-

<%@ Page Title="" Language="C#" MasterPageFile="~/Site1.Master" AutoEventWireup="true" CodeBehind="adminbookissuing.aspx.cs" Inherits="E\_library.adminbookissuing" %>

<asp:Content ID="Content1" ContentPlaceHolderID="head" runat="server">

<link href="Content/adminbookissuing/StyleSheet1.css" rel="stylesheet" />

<script type="text/javascript">

$(document).ready(function () {

$(".table").prepend($("

< thead ></thead > ").append($(this).find("tr:first"))).dataTable();

});

</script>

</asp:Content>

<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1" runat="server">

<div class="container-fluid">

<div class="row">

<div class="col-md-5">

<div class="card">

<div class="card-body">

<div class="row">

<div class="col">

<center>

<h4>Book Issuing</h4>

</center>

</div>

</div>

<div class="row">

<div class="col">

<center>

<img src="Images/books.png" width="100px" />

</center>

</div>

</div>

<div class="row">

<div class="col">

<hr>

</div>

</div>

<div class="row">

<div class="col-md-6">

<label>Member ID</label>

<div class="form-group">

<asp:TextBox CssClass="form-control" ID="TextBox2" runat="server" placeholder="Member ID"></asp:TextBox>

</div>

</div>

<div class="col-md-6">

<label>Book ID</label>

<div class="form-group">

<div class="input-group">

<asp:TextBox CssClass="form-control" ID="TextBox1" runat="server" placeholder="Book ID"></asp:TextBox>

<asp:Button for="TextBox1" class="btn btn-dark" ID="Button1" runat="server" Text="Go" OnClick="Button1\_Click" />

</div>

</div>

</div>

</div>

<div class="row">

<div class="col-md-6">

<label>Member Name</label>

<div class="form-group">

<asp:TextBox CssClass="form-control" ID="TextBox3" runat="server" placeholder="Member Name" ReadOnly="True"></asp:TextBox>

</div>

</div>

<div class="col-md-6">

<label>Book Name</label>

<div class="form-group">

<asp:TextBox CssClass="form-control" ID="TextBox4" runat="server" placeholder="Book Name" ReadOnly="True"></asp:TextBox>

</div>

</div>

</div>

<div class="row">

<div class="col-md-6">

<label>Issue Date</label>

<div class="form-group">

<asp:TextBox CssClass="form-control" ID="TextBox5" runat="server" placeholder="Start Date" TextMode="Date"></asp:TextBox>

</div>

</div>

<div class="col-md-6">

<label>DUe Date</label>

<div class="form-group">

<asp:TextBox CssClass="form-control" ID="TextBox6" runat="server" placeholder="End Date" TextMode="Date"></asp:TextBox>

</div>

</div>

</div>

<div class="row">

<div class="col-6">

<asp:Button ID="Button2" OnClick="Button2\_Click" class="btn btn-lg btn-block btn-primary" runat="server" Text="Issue" />

</div>

<div class="col-6">

<asp:Button ID="Button4" OnClick="Button4\_Click" class="btn btn-lg btn-block btn-success" runat="server" Text="Return" />

</div>

</div>

</div>

</div>

<a href="homepage.aspx"><< Back to Home</a><br>

<br>

</div>

<div class="col-md-7">

<div class="card">

<div class="card-body">

<div class="row">

<div class="col">

<center>

<h4>Issued Book List</h4>

</center>

</div>

</div>

<div class="row">

<div class="col">

<hr>

</div>

</div>

<div class="row">

<asp:SqlDataSource ID="SqlDataSource1" runat="server" ConnectionString='<%$ ConnectionStrings:elibraryDBConnectionString %>' SelectCommand="SELECT \* FROM [book\_issue\_tbl]">

</asp:SqlDataSource>

<div class="col">

<asp:GridView class="table table-striped table-bordered" ID="GridView1" runat="server" AutoGenerateColumns="False" DataSourceID="SqlDataSource1" OnRowDataBound="GridView1\_RowDataBound" >

<Columns>

<asp:BoundField DataField="member\_id" HeaderText="Member ID" SortExpression="member\_id"></asp:BoundField>

<asp:BoundField DataField="member\_name" HeaderText="Member Name" SortExpression="member\_name"></asp:BoundField>

<asp:BoundField DataField="book\_id" HeaderText="Book ID" SortExpression="book\_id"></asp:BoundField>

<asp:BoundField DataField="book\_name" HeaderText="Book Name" SortExpression="book\_name"></asp:BoundField>

<asp:BoundField DataField="issue\_date" HeaderText="Issue Date" SortExpression="issue\_date"></asp:BoundField>

<asp:BoundField DataField="due\_date" HeaderText="Due Date" SortExpression="due\_date"></asp:BoundField>

</Columns>

</asp:GridView>

</div>

</div>

</div>

</div>

</div>

</div>

</div>

</asp:Content>

* + 1. **Code For Book Issueing:-**

1. using System;
2. using System.Collections.Generic;
3. using System.Configuration;
4. using System.Data.SqlClient;
5. using System.Data;
6. using System.Linq;
7. using System.Web;
8. using System.Web.UI;
9. using System.Web.UI.WebControls;
10. namespace E\_library
11. {
12. public partial class adminbookissuing : System.Web.UI.Page
13. {
14. string strcon = ConfigurationManager.ConnectionStrings["con"].ConnectionString;
15. protected void Page\_Load(object sender, EventArgs e)
16. {
17. GridView1.DataBind();
18. }
19. //go button click event
20. protected void Button1\_Click(object sender, EventArgs e)
21. {
22. getNames();
23. }
24. //issue book
25. protected void Button2\_Click(object sender, EventArgs e)
26. {
27. if (checkIfBookExist() && checkIfMemberExist())
28. {
29. if (checkIfIssueEntryExist())
30. {
31. Response.Write("<script>alert('This Member already has this book');</script>");
32. }
33. else
34. {
35. issueBook();
36. }
37. }
38. else
39. {
40. Response.Write("<script>alert('Wrong Book ID or Member ID');</script>");
41. }
42. }
43. //return book
44. protected void Button4\_Click(object sender, EventArgs e)
45. {
46. if (checkIfBookExist() && checkIfMemberExist())
47. {
48. if (checkIfIssueEntryExist())
49. {
50. returnBook();
51. }
52. else
53. {
54. Response.Write("<script>alert('This Entry does not exist');</script>");
55. }
56. }
57. else
58. {
59. Response.Write("<script>alert('Wrong Book ID or Member ID');</script>");
60. }
61. }
62. // go button click event
64. // user defined function
65. void returnBook()
66. {
67. try
68. {
69. SqlConnection con = new SqlConnection(strcon);
70. if (con.State == ConnectionState.Closed)
71. {
72. con.Open();
73. }
74. SqlCommand cmd = new SqlCommand("Delete from book\_issue\_tbl WHERE book\_id='" + TextBox1.Text.Trim() + "' AND member\_id='" + TextBox2.Text.Trim() + "'", con);
75. int result = cmd.ExecuteNonQuery();
76. if (result > 0)
77. {
78. cmd = new SqlCommand("update book\_master\_tbl set current\_stock = current\_stock+1 WHERE book\_id='" + TextBox1.Text.Trim() + "'", con);
79. cmd.ExecuteNonQuery();
80. con.Close();
81. Response.Write("<script>alert('Book Returned Successfully');</script>");
82. GridView1.DataBind();
83. con.Close();
84. }
85. else
86. {
87. Response.Write("<script>alert('Error - Invalid details');</script>");
88. }
89. }
90. catch (Exception ex)
91. {
92. Response.Write("<script>alert('" + ex.Message + "');</script>");
93. }
94. }
95. void issueBook()
96. {
97. try
98. {
99. SqlConnection con = new SqlConnection(strcon);
100. if (con.State == ConnectionState.Closed)
101. {
102. con.Open();
103. }
104. SqlCommand cmd = new SqlCommand("INSERT INTO book\_issue\_tbl(member\_id,member\_name,book\_id,book\_name,issue\_date,due\_date) values(@member\_id,@member\_name,@book\_id,@book\_name,@issue\_date,@due\_date)", con);
105. cmd.Parameters.AddWithValue("@member\_id", TextBox2.Text.Trim());
106. cmd.Parameters.AddWithValue("@member\_name", TextBox3.Text.Trim());
107. cmd.Parameters.AddWithValue("@book\_id", TextBox1.Text.Trim());
108. cmd.Parameters.AddWithValue("@book\_name", TextBox4.Text.Trim());
109. cmd.Parameters.AddWithValue("@issue\_date", TextBox5.Text.Trim());
110. cmd.Parameters.AddWithValue("@due\_date", TextBox6.Text.Trim());
111. cmd.ExecuteNonQuery();
112. cmd = new SqlCommand("update book\_master\_tbl set current\_stock = current\_stock-1 WHERE book\_id='" + TextBox1.Text.Trim() + "'", con);
113. cmd.ExecuteNonQuery();
114. con.Close();
115. Response.Write("<script>alert('Book Issued Successfully');</script>");
116. GridView1.DataBind();
117. }
118. catch (Exception ex)
119. {
120. Response.Write("<script>alert('" + ex.Message + "');</script>");
121. }
122. }
123. bool checkIfBookExist()
124. {
125. try
126. {
127. SqlConnection con = new SqlConnection(strcon);
128. if (con.State == ConnectionState.Closed)
129. {
130. con.Open();
131. }
132. SqlCommand cmd = new SqlCommand("select \* from book\_master\_tbl WHERE book\_id='" + TextBox1.Text.Trim() + "' AND current\_stock >0", con);
133. SqlDataAdapter da = new SqlDataAdapter(cmd);
134. DataTable dt = new DataTable();
135. da.Fill(dt);
136. if (dt.Rows.Count >= 1)
137. {
138. return true;
139. }
140. else
141. {
142. return false;
143. }
144. }
145. catch (Exception ex)
146. {
147. return false;
148. }
149. }
150. bool checkIfMemberExist()
151. {
152. try
153. {
154. SqlConnection con = new SqlConnection(strcon);
155. if (con.State == ConnectionState.Closed)
156. {
157. con.Open();
158. }
159. SqlCommand cmd = new SqlCommand("select full\_name from member\_master\_tbl WHERE member\_id='" + TextBox2.Text.Trim() + "'", con);
160. SqlDataAdapter da = new SqlDataAdapter(cmd);
161. DataTable dt = new DataTable();
162. da.Fill(dt);
163. if (dt.Rows.Count >= 1)
164. {
165. return true;
166. }
167. else
168. {
169. return false;
170. }
171. }
172. catch (Exception ex)
173. {
174. return false;
175. }
176. }
177. bool checkIfIssueEntryExist()
178. {
179. try
180. {
181. SqlConnection con = new SqlConnection(strcon);
182. if (con.State == ConnectionState.Closed)
183. {
184. con.Open();
185. }
186. SqlCommand cmd = new SqlCommand("select \* from book\_issue\_tbl WHERE member\_id='" + TextBox2.Text.Trim() + "' AND book\_id='" + TextBox1.Text.Trim() + "'", con);
187. SqlDataAdapter da = new SqlDataAdapter(cmd);
188. DataTable dt = new DataTable();
189. da.Fill(dt);
190. if (dt.Rows.Count >= 1)
191. {
192. return true;
193. }
194. else
195. {
196. return false;
197. }
198. }
199. catch (Exception ex)
200. {
201. return false;
202. }
203. }
204. void getNames()
205. {
206. try
207. {
208. SqlConnection con = new SqlConnection(strcon);
209. if (con.State == ConnectionState.Closed)
210. {
211. con.Open();
212. }
213. SqlCommand cmd = new SqlCommand("select book\_name from book\_master\_tbl WHERE book\_id='" + TextBox1.Text.Trim() + "'", con);
214. SqlDataAdapter da = new SqlDataAdapter(cmd);
215. DataTable dt = new DataTable();
216. da.Fill(dt);
217. if (dt.Rows.Count >= 1)
218. {
219. TextBox4.Text = dt.Rows[0]["book\_name"].ToString();
220. }
221. else
222. {
223. Response.Write("<script>alert('Wrong Book ID');</script>");
224. }
225. cmd = new SqlCommand("select full\_name from member\_master\_tbl WHERE member\_id='" + TextBox2.Text.Trim() + "'", con);
226. da = new SqlDataAdapter(cmd);
227. dt = new DataTable();
228. da.Fill(dt);
229. if (dt.Rows.Count >= 1)
230. {
231. TextBox3.Text = dt.Rows[0]["full\_name"].ToString();
232. }
233. else
234. {
235. Response.Write("<script>alert('Wrong User ID');</script>");
236. }
237. }
238. catch (Exception ex)
239. {
240. }
241. }
242. protected void GridView1\_RowDataBound(object sender, GridViewRowEventArgs e)
243. {
244. try
245. {
246. if (e.Row.RowType == DataControlRowType.DataRow)
247. {
248. //Check your condition here
249. DateTime dt = Convert.ToDateTime(e.Row.Cells[5].Text);
250. DateTime today = DateTime.Today;
251. if (today > dt)
252. {
253. e.Row.BackColor = System.Drawing.Color.PaleVioletRed;
254. }
255. }
256. }
257. catch (Exception ex)
258. {
259. Response.Write("<script>alert('" + ex.Message + "');</script>");
260. }
261. }
262. }
263. }

**4.1.7 Member/Student Management:-**

<%@ Page Title="" Language="C#" MasterPageFile="~/Site1.Master" AutoEventWireup="true" CodeBehind="adminmembermanagement.aspx.cs" Inherits="E\_library.adminmembermanagement" %>

<asp:Content ID="Content1" ContentPlaceHolderID="head" runat="server">

<link href="Content/adminmembermanagement/StyleSheet1.css" rel="stylesheet" />

<script type="text/javascript">

$(document).ready(function () {

$(".table").prepend($("<thead></thead>").append($(this).find("tr:first"))).dataTable();

});

</script>>

</asp:Content>

<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1" runat="server">

<div class="container-fluid">

<div class="row">

<div class="col-md-5">

<div class="card">

<div class="card-body">

<div class="row">

<div class="col">

<center>

<h4>Member Details</h4>

</center>

</div>

</div>

<div class="row">

<div class="col">

<center>

<img src="Images/generaluser.png" width="100px"/>

</center>

</div>

</div>

<div class="row">

<div class="col">

<hr>

</div>

</div>

<div class="row">

<div class="col-md-3">

<label>Member ID</label>

<div class="form-group">

<div class="input-group">

<asp:TextBox CssClass="form-control" ID="TextBox1" runat="server" placeholder="Member ID"></asp:TextBox>

<asp:LinkButton class="btn btn-primary" ID="LinkButton4" runat="server" OnClick="LinkButton4\_Click"><i class="fas fa-check-circle"></i></asp:LinkButton>

</div>

</div>

</div>

<div class="col-md-4">

<label>Full Name</label>

<div class="form-group">

<asp:TextBox CssClass="form-control" ID="TextBox2" runat="server" placeholder="Full Name" ReadOnly="True"></asp:TextBox>

</div>

</div>

<div class="col-md-5">

<label>Account Status</label>

<div class="form-group">

<div class="input-group">

<asp:TextBox CssClass="form-control mr-1" ID="TextBox7" runat="server" placeholder="Account Status" ReadOnly="True"></asp:TextBox>

<asp:LinkButton class="btn btn-success mr-1" ID="LinkButton1" runat="server" OnClick="LinkButton1\_Click"><i class="fas fa-check-circle"></i></asp:LinkButton>

<asp:LinkButton class="btn btn-warning mr-1" ID="LinkButton2" runat="server" OnClick="LinkButton2\_Click"><i class="far fa-pause-circle"></i></asp:LinkButton>

<asp:LinkButton class="btn btn-danger mr-1" ID="LinkButton3" runat="server" OnClick="LinkButton3\_Click"><i class="fas fa-times-circle"></i></asp:LinkButton>

</div>

</div>

</div>

</div>

<div class="row">

<div class="col-md-3">

<label>DOB</label>

<div class="form-group">

<asp:TextBox CssClass="form-control" ID="TextBox8" runat="server" placeholder="DOB" ReadOnly="True"></asp:TextBox>

</div>

</div>

<div class="col-md-4">

<label>Contact No</label>

<div class="form-group">

<asp:TextBox CssClass="form-control" ID="TextBox3" runat="server" placeholder="Contact No" ReadOnly="True"></asp:TextBox>

</div>

</div>

<div class="col-md-5">

<label>Email ID</label>

<div class="form-group">

<asp:TextBox CssClass="form-control" ID="TextBox4" runat="server" placeholder="Email ID" ReadOnly="True"></asp:TextBox>

</div>

</div>

</div>

<div class="row">

<div class="col-md-4">

<label>State</label>

<div class="form-group">

<asp:TextBox CssClass="form-control" ID="TextBox9" runat="server" placeholder="State" ReadOnly="True"></asp:TextBox>

</div>

</div>

<div class="col-md-4">

<label>City</label>

<div class="form-group">

<asp:TextBox CssClass="form-control" ID="TextBox10" runat="server" placeholder="City" ReadOnly="True"></asp:TextBox>

</div>

</div>

<div class="col-md-4">

<label>Pin Code</label>

<div class="form-group">

<asp:TextBox CssClass="form-control" ID="TextBox11" runat="server" placeholder="Pin Code" ReadOnly="True"></asp:TextBox>

</div>

</div>

</div>

<div class="row">

<div class="col-12">

<label>Full Postal Address</label>

<div class="form-group">

<asp:TextBox CssClass="form-control" ID="TextBox6" runat="server" placeholder="Full Postal Address" TextMode="MultiLine" Rows="2" ReadOnly="True"></asp:TextBox>

</div>

</div>

</div>

<div class="row">

<div class="col-8 mx-auto">

<asp:Button ID="Button2" class="btn btn-lg btn-block btn-danger" runat="server" Text="Delete User Permanently" OnClick="Button2\_Click" />

</div>

</div>

</div>

</div>

<a href="homepage.aspx" style="background-color:whitesmoke;background-blend-mode:lighten"><< Back to Home</a><br>

<br>

</div>

<div class="col-md-7">

<div class="card">

<div class="card-body">

<div class="row">

<div class="col">

<center>

<h4>Member List</h4>

</center>

</div>

</div>

<div class="row">

<div class="col">

<hr>

</div>

</div>

<div class="row">

<asp:SqlDataSource ID="SqlDataSource1" runat="server" ConnectionString="<%$ ConnectionStrings:elibraryDBConnectionString %>" SelectCommand="SELECT \* FROM [member\_master\_tbl]"></asp:SqlDataSource>

<div class="col">

<asp:GridView class="table table-striped table-bordered" ID="GridView1" runat="server" AutoGenerateColumns="False" DataKeyNames="member\_id" DataSourceID="SqlDataSource1" >

<Columns>

<asp:BoundField DataField="member\_id" HeaderText="ID" ReadOnly="True" SortExpression="member\_id" />

<asp:BoundField DataField="full\_name" HeaderText="Name" SortExpression="full\_name" />

<asp:BoundField DataField="account\_status" HeaderText="Account Status" SortExpression="account\_status" />

<asp:BoundField DataField="contact\_no" HeaderText="Contact No" SortExpression="contact\_no" />

<asp:BoundField DataField="email" HeaderText="Email ID" SortExpression="email" />

<asp:BoundField DataField="state" HeaderText="State" SortExpression="state" />

<asp:BoundField DataField="city" HeaderText="City" SortExpression="city" />

<asp:BoundField DataField="pincode" HeaderText="pincode" SortExpression="pincode" />

</Columns>

</asp:GridView>

</div>

</div>

</div>

</div>

</div>

</div>

</div>

</asp:Content>

**4.1.7.1 Code Of Member/Student Management:-**

using System;

using System.Collections.Generic;

using System.Configuration;

using System.Data;

using System.Data.SqlClient;

using System.Linq;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

namespace E\_library

{

public partial class adminmembermanagement : System.Web.UI.Page

{

string strcon = ConfigurationManager.ConnectionStrings["con"].ConnectionString;

protected void Page\_Load(object sender, EventArgs e)

{

GridView1.DataBind();

}

//link button 4 or go button

protected void LinkButton4\_Click(object sender, EventArgs e)

{

getMemberById();

}

//green button or

protected void LinkButton1\_Click(object sender, EventArgs e)

{

updateMemberStatusByID("active");

}

//Onhold or yellow button

protected void LinkButton2\_Click(object sender, EventArgs e)

{

updateMemberStatusByID("Pending");

}

//block or banned account

protected void LinkButton3\_Click(object sender, EventArgs e)

{

updateMemberStatusByID("deactive");

}

protected void Button2\_Click(object sender, EventArgs e)

{

deleteMemberByID();

}

//user defined function

bool checkIfMemberExists()

{

try

{

SqlConnection con = new SqlConnection(strcon);

if (con.State == ConnectionState.Closed)

{

con.Open();

}

SqlCommand cmd = new SqlCommand("SELECT \* from member\_master\_tbl where member\_id='" + TextBox1.Text.Trim() + "';", con);

SqlDataAdapter da = new SqlDataAdapter(cmd);

DataTable dt = new DataTable();

da.Fill(dt);

if (dt.Rows.Count >= 1)

{

return true;

}

else

{

return false;

}

}

catch (Exception ex)

{

Response.Write("<script>alert('" + ex.Message + "');</script>");

return false;

}

}

//delete member id

void deleteMemberByID()

{

if (checkIfMemberExists())

{

try

{

SqlConnection con = new SqlConnection(strcon);

if (con.State == ConnectionState.Closed)

{

con.Open();

}

SqlCommand cmd = new SqlCommand("DELETE from member\_master\_tbl WHERE member\_id='" + TextBox1.Text.Trim() + "'", con);

cmd.ExecuteNonQuery();

con.Close();

Response.Write("<script>alert('Member Deleted Successfully');</script>");

clearForm();

GridView1.DataBind();

}

catch (Exception ex)

{

Response.Write("<script>alert('" + ex.Message + "');</script>");

}

}

else

{

Response.Write("<script>alert('Invalid Member ID');</script>");

}

}

void getMemberById()

{

try

{

SqlConnection con = new SqlConnection(strcon);

if (con.State == ConnectionState.Closed)

{

con.Open();

}

SqlCommand cmd = new SqlCommand("Select \* from member\_master\_tbl where member\_id ='" + TextBox1.Text.Trim() + "' ", con);

SqlDataReader dr = cmd.ExecuteReader();

if (dr.HasRows)

{

while (dr.Read())

{

TextBox2.Text = dr.GetValue(0).ToString(); //full name

TextBox7.Text = dr.GetValue(10).ToString(); //account status

TextBox8.Text = dr.GetValue(1).ToString(); //dob

TextBox3.Text = dr.GetValue(2).ToString(); //contact no

TextBox4.Text = dr.GetValue(3).ToString(); //emailID

TextBox9.Text = dr.GetValue(4).ToString(); //State

TextBox10.Text = dr.GetValue(5).ToString(); //City

TextBox11.Text = dr.GetValue(6).ToString(); //pincode

TextBox6.Text = dr.GetValue(7).ToString();//Full address

/\* Response.Write("<script>alert('" + dr.GetValue(0).ToString() + "')</script>");

Session["username"] = dr.GetValue(0).ToString();

Session["fullname"] = dr.GetValue(2).ToString();

Session["role"] = "admin";

//Session["status"] = dr.GetValue(10).ToString();\*/

}

}

else

{

Response.Write("<script>alert('This Member ID has not been Registered Yet .. ');</script>");

}

}

catch (Exception ex)

{

Response.Write("<script>alert('" + ex.Message + "')</script>");

}

}

void updateMemberStatusByID(string status)

{

if (checkIfMemberExists())

{

try

{

SqlConnection con = new SqlConnection(strcon);

if (con.State == ConnectionState.Closed)

{

con.Open();

}

SqlCommand cmd = new SqlCommand("UPDATE member\_master\_tbl SET account\_status='" + status + "' WHERE member\_id='" + TextBox1.Text.Trim() + "'", con);

cmd.ExecuteNonQuery();

con.Close();

GridView1.DataBind();

Response.Write("<script>alert('Member Status Updated');</script>");

}

catch (Exception ex)

{

Response.Write("<script>alert('" + ex.Message + "');</script>");

}

}

else

{

Response.Write("<script>alert('Invalid Member ID');</script>");

}

}

void clearForm()

{

TextBox1.Text = "";

TextBox2.Text = "";

TextBox7.Text = "";

TextBox8.Text = "";

TextBox3.Text = "";

TextBox4.Text = "";

TextBox9.Text = "";

TextBox10.Text = "";

TextBox11.Text = "";

TextBox6.Text = "";

}

}

}

**4.1.8Code for Publisher Management:-**

using System;

using System.Collections.Generic;

using System.Configuration;

using System.Data;

using System.Data.SqlClient;

using System.Linq;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

namespace E\_library

{

public partial class aadminpublishermanagement : System.Web.UI.Page

{

String strcon = ConfigurationManager.ConnectionStrings["con"].ConnectionString;

protected void Page\_Load(object sender, EventArgs e)

{

GridView1.DataBind();

}

//Add new publisher :-

protected void Button2\_Click(object sender, EventArgs e)

{

if (checkIfPublisherExists())

{

Response.Write("<script>alert('Publisher with this ID already exists... you cannot add another Publisher with same Publisher ID ');</script>");

}

else

{

AddNewPublisher();

}

}

//update Button Click

protected void Button3\_Click(object sender, EventArgs e)

{

if (checkIfPublisherExists())

{

updatePublisher();

}

else

{

Response.Write("<script>alert('Publisher Does Not Exist... ');</script>");

}

}

//Delete Button Click

protected void Button4\_Click(object sender, EventArgs e)

{

if (checkIfPublisherExists())

{

DeletePublisher();

}

else

{

Response.Write("<script>alert('Publisher Does Not Exist... ');</script>");

}

}

//Go button click

protected void Button1\_Click(object sender, EventArgs e)

{

getPublisherById();

}

//User DEfined Function

//go buttom

void getPublisherById()

{

try

{

SqlConnection con = new SqlConnection(strcon);

if (con.State == ConnectionState.Closed)

{

con.Open();

}

SqlCommand cmd = new SqlCommand("SELECT \* from publisher\_master\_tbl where publisher\_id='" + TextBox1.Text.Trim() + "'", con);

SqlDataAdapter da = new SqlDataAdapter(cmd);

DataTable dt = new DataTable();

da.Fill(dt);

if (dt.Rows.Count >= 1)

{

TextBox2.Text = dt.Rows[0][1].ToString();

}

else

{

Response.Write("<script>alert('INVALID PUBLISHER ID')</script>");

}

}

catch (Exception ex)

{

Response.Write("<script>alert('" + ex.Message + "');</script>");

}

}

//to delete Publisher

void DeletePublisher()

{

try

{

SqlConnection con = new SqlConnection(strcon);

if (con.State == ConnectionState.Closed)

{

con.Open();

}

SqlCommand cmd = new SqlCommand("DELETE from publisher\_master\_tbl WHERE publisher\_id='" + TextBox1.Text.Trim() + "'", con);

cmd.ExecuteNonQuery();

con.Close();

Response.Write("<script>alert('Publisher Deleted Successfully ');</script>");

clearForm();

GridView1.DataBind();

}

catch (Exception ex)

{

Response.Write("<script>alert('" + ex.Message + "');</script>");

}

}

//for Updating Publisher detail/Name

void updatePublisher()

{

try

{

SqlConnection con = new SqlConnection(strcon);

if (con.State == ConnectionState.Closed)

{

con.Open();

}

SqlCommand cmd = new SqlCommand("UPDATE publisher\_master\_tbl publisher\_name = @publisher\_name WHERE publisher\_id='" + TextBox1.Text.Trim()+"'", con);

//basically values are parameter which can be anthing but use same

cmd.Parameters.AddWithValue("@publisher\_name", TextBox2.Text.Trim());

cmd.ExecuteNonQuery();

con.Close();

Response.Write("<script>alert('Publisher Updated Successfully ');</script>");

clearForm();

GridView1.DataBind();

}

catch (Exception ex)

{

Response.Write("<script>alert('" + ex.Message + "');</script>");

}

}

//for adding new author

void AddNewPublisher()

{

try

{

SqlConnection con = new SqlConnection(strcon);

if (con.State == ConnectionState.Closed)

{

con.Open();

}

SqlCommand cmd = new SqlCommand("INSERT INTO publisher\_master\_tbl(publisher\_id,publisher\_name) values(@publisher\_id,@publisher\_name) ", con);

cmd.Parameters.AddWithValue("@publisher\_id", TextBox1.Text.Trim()); //basically values are parameter which can be anthing but use same

cmd.Parameters.AddWithValue("@publisher\_name", TextBox2.Text.Trim());

cmd.ExecuteNonQuery();

con.Close();

Response.Write("<script>alert('Publisher added Successfully ');</script>");

clearForm();

GridView1.DataBind();

}

catch (Exception ex)

{

Response.Write("<script>alert('" + ex.Message + "');</script>");

}

}

//for chechking if Author is present or not

bool checkIfPublisherExists()

{

try

{

SqlConnection con = new SqlConnection(strcon);

if (con.State == ConnectionState.Closed)

{

con.Open();

}

SqlCommand cmd = new SqlCommand("SELECT \* from publisher\_master\_tbl where publisher\_id='" + TextBox1.Text.Trim() + "'", con);

SqlDataAdapter da = new SqlDataAdapter(cmd);

DataTable dt = new DataTable();

da.Fill(dt);

if (dt.Rows.Count >= 1)

{

return true;

}

else

{

return false;

}

}

catch (Exception ex)

{

Response.Write("<script>alert('" + ex.Message + "');</script>");

return false;

}

}

void clearForm()

{

TextBox1.Text = "";

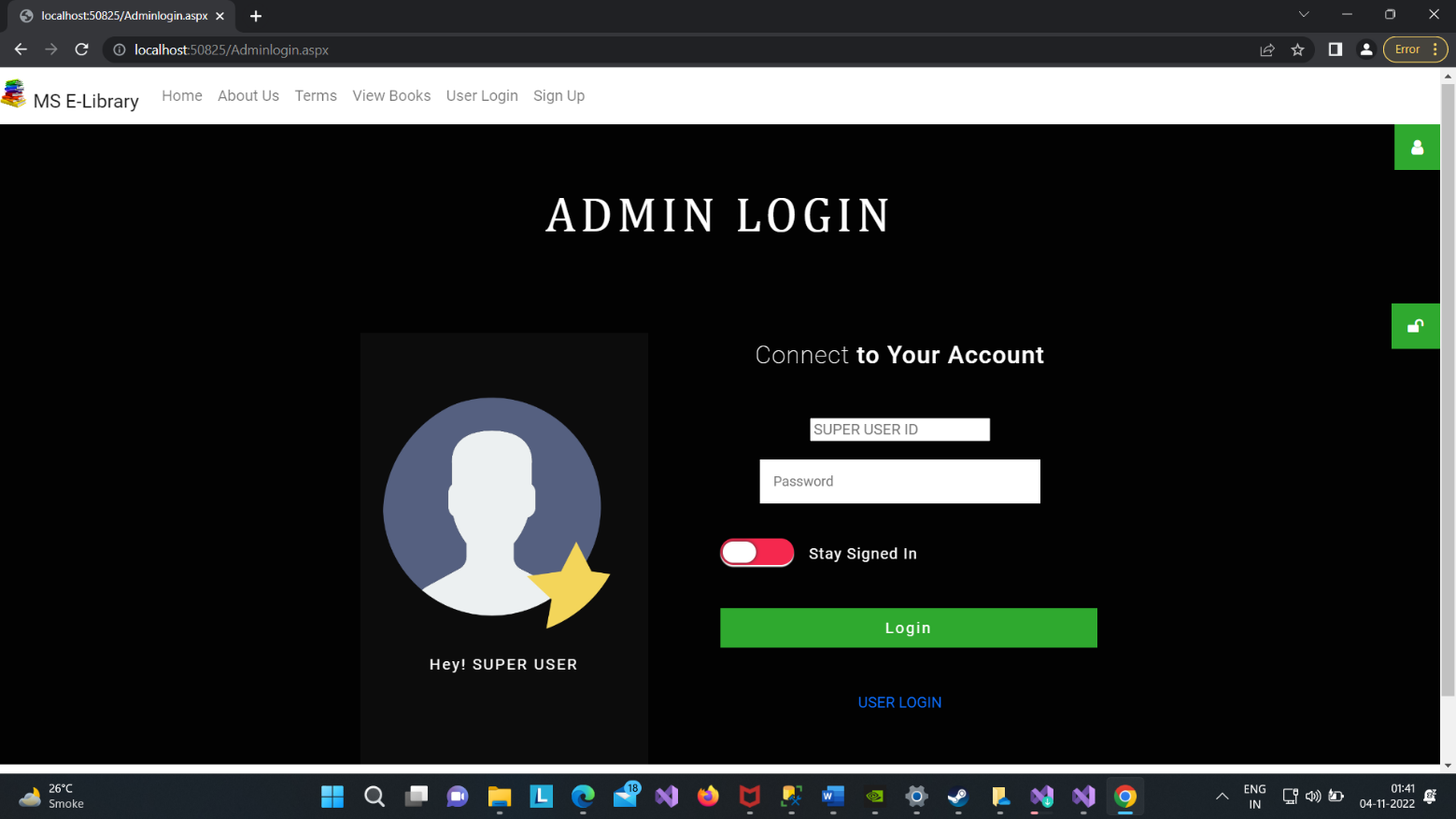
TextBox2.Text = "";

}

}

}

* + 1. **Screenshot for Admin login**



#### Student Module

STUDENT

ONLINE BOOK RESERVATTION

EVENT ADDITION

STUDENT LOGIN

FORGOT PASSWORD

USERNAME AND PASSWORD

AUTHENTICATION

STUDENT REGISTRATION

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.

#### Code For Student account creation

<%@ Page Title="" Language="C#" MasterPageFile="~/Site1.Master" AutoEventWireup="true" CodeBehind="Signup.aspx.cs" Inherits="E\_library.Signup" %>

<asp:Content ID="Content1" ContentPlaceHolderID="head" runat="server">

<style>

body{

background-image:url( '../../Images/lib\_login.jpg');

}

</style>

</asp:Content>

<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1" runat="server">

<div class="container">

<div class="row">

<div class="col-md-8 mx-auto">

<div class="card">

<div class="card-body">

<div class="row">

<div class="col">

<center>

<img width="100px"src="Images/generaluser.png" />

</center>

</div>

</div>

<div class="row">

<div class="col">

<center>

<h4>USER/MEMBER SIGNUP</h4>

</center>

</div>

</div>

<div class="row">

<div class="col">

<hr>

</div>

</div>

<div class="row">

<div class="col-md-6">

<label>Full Name</label>

<div class="form-group">

<asp:TextBox CssClass="form-control" ID="TextBox1" runat="server" placeholder="Full Name"></asp:TextBox>

</div>

</div>

<div class="col-md-6">

<label>Date Of Birth</label>

<div class="form-group">

<asp:TextBox CssClass="form-control" ID="TextBox2" runat="server" placeholder="Password" TextMode="Date"></asp:TextBox>

</div>

</div>

</div>

<div class="row">

<div class="col-md-6">

<label>Contact no</label>

<div class="form-group">

<asp:TextBox CssClass="form-control" ID="TextBox3" runat="server" placeholder="Contact No" TextMode="Number"></asp:TextBox>

</div>

</div>

<div class="col-md-6">

<label>Email</label>

<div class="form-group">

<asp:TextBox CssClass="form-control" ID="TextBox4" runat="server" placeholder="Email" TextMode="Email"></asp:TextBox>

</div>

</div>

</div>

<div class="row">

<div class="col-md-4">

<label>State</label>

<div class="form-group">

<asp:DropDownList class="form-control" ID="DropDownList" runat="server" >

<asp:ListItem Text="Select" Value="Select">

</asp:ListItem>

<asp:ListItem Text=" Andhra Pradesh" Value=" Andhra Pradesh">

</asp:ListItem>

<asp:ListItem Text="Arunachal Pradesh" Value="Arunachal Pradesh">

</asp:ListItem>

<asp:ListItem Text="Assam" Value="Assam">

</asp:ListItem>

<asp:ListItem Text="Bihar" Value="Bihar">

</asp:ListItem>

<asp:ListItem Text="Chhattisgarh" Value="Chhattisgarh">

</asp:ListItem>

<asp:ListItem Text="Goa " Value="Goa ">

</asp:ListItem>

<asp:ListItem Text="Gujarat" Value="Gujarat">

</asp:ListItem>

<asp:ListItem Text="Haryana" Value="Haryana">

</asp:ListItem>

<asp:ListItem Text="Himachal Pradesh" Value="Himachal Pradesh">

</asp:ListItem>

<asp:ListItem Text="Jammu and Kashmi" Value="Jammu and Kashmi">

</asp:ListItem>

<asp:ListItem Text="Jharkhand" Value="Jharkhand">

</asp:ListItem>

<asp:ListItem Text="Karnataka" Value="Karnataka">

</asp:ListItem>

<asp:ListItem Text="Kerala" Value="Kerala">

</asp:ListItem>

<asp:ListItem Text="Maharashtra" Value="Maharashtra">

</asp:ListItem>

<asp:ListItem Text="Manipur" Value="Manipur">

</asp:ListItem>

<asp:ListItem Text="Meghalaya" Value="Meghalaya">

</asp:ListItem>

<asp:ListItem Text="Mizoram" Value="Mizoram">

</asp:ListItem>

<asp:ListItem Text="Nagaland" Value="Nagaland">

</asp:ListItem>

<asp:ListItem Text="Odisha" Value="Odisha">

</asp:ListItem>

<asp:ListItem Text="Punjab " Value="Punjab ">

</asp:ListItem>

<asp:ListItem Text="Rajasthan" Value="Rajasthan">

</asp:ListItem>

<asp:ListItem Text="Sikkim" Value="Sikkim">

</asp:ListItem>

<asp:ListItem Text="Tamil Nadu" Value="Tamil Nadu">

</asp:ListItem>

<asp:ListItem Text="Telangana " Value="Telangana ">

</asp:ListItem>

<asp:ListItem Text="Tripura" Value="Tripura">

</asp:ListItem>

<asp:ListItem Text="Uttarakhand" Value="Uttarakhand">

</asp:ListItem>

<asp:ListItem Text="Uttar Pradesh" Value="Uttar Pradesh">

</asp:ListItem>

<asp:ListItem Text="West Bengal" Value="West Bengal">

</asp:ListItem>

</asp:DropDownList>

</div>

</div>

<div class="col-md-4">

<label>City</label>

<div class="form-group">

<asp:TextBox CssClass="form-control" ID="TextBox5" runat="server" placeholder="City" TextMode="SingleLine"></asp:TextBox>

</div>

</div>

<div class="col-md-4">

<label>Pin-code</label>

<div class="form-group">

<asp:TextBox CssClass="form-control" ID="TextBox6" runat="server" placeholder="Pincode" TextMode="Number"></asp:TextBox>

</div>

</div>

</div>

<div class="row">

<div class="col">

<label>Full Address</label>

<div class="form-group">

<asp:TextBox CssClass="form-control" ID="TextBox7" runat="server" placeholder="Full Address" TextMode="MultiLine" Rows="2"></asp:TextBox>

</div>

<div class="row">

<div class="col">

<center>

<span class="badge rounded-pill bg-info text-dark">Login Credentials</span>

</center>

</div>

</div>

<br />

<div class="row">

<div class="col-md-6">

<label>USER/MEMBER ID</label>

<div class="form-group">

<asp:TextBox CssClass="form-control" ID="TextBox8" runat="server" placeholder="USER/MEMBER ID" TextMode="SingleLine"></asp:TextBox>

</div>

</div>

<div class="col-md-6">

<label>Password</label>

<div class="form-group">

<asp:TextBox CssClass="form-control" ID="TextBox9" runat="server" placeholder="Password" TextMode="Password"></asp:TextBox>

</div>

</div>

</div>

<br />

</div>

<center>

<div class="row">

<div class="col">

<div class="form-group">

<asp:Button class="btn btn-success btn-block btn-lg" ID="Button1" runat="server" Text="Sign Up" OnClick="Button1\_Click" Width="171px" />

</div>

</div>

</div>

</center>

</div>

</div>

</div>

<a href="homepage.aspx"><< Back to Home</a><br><br>

</div>

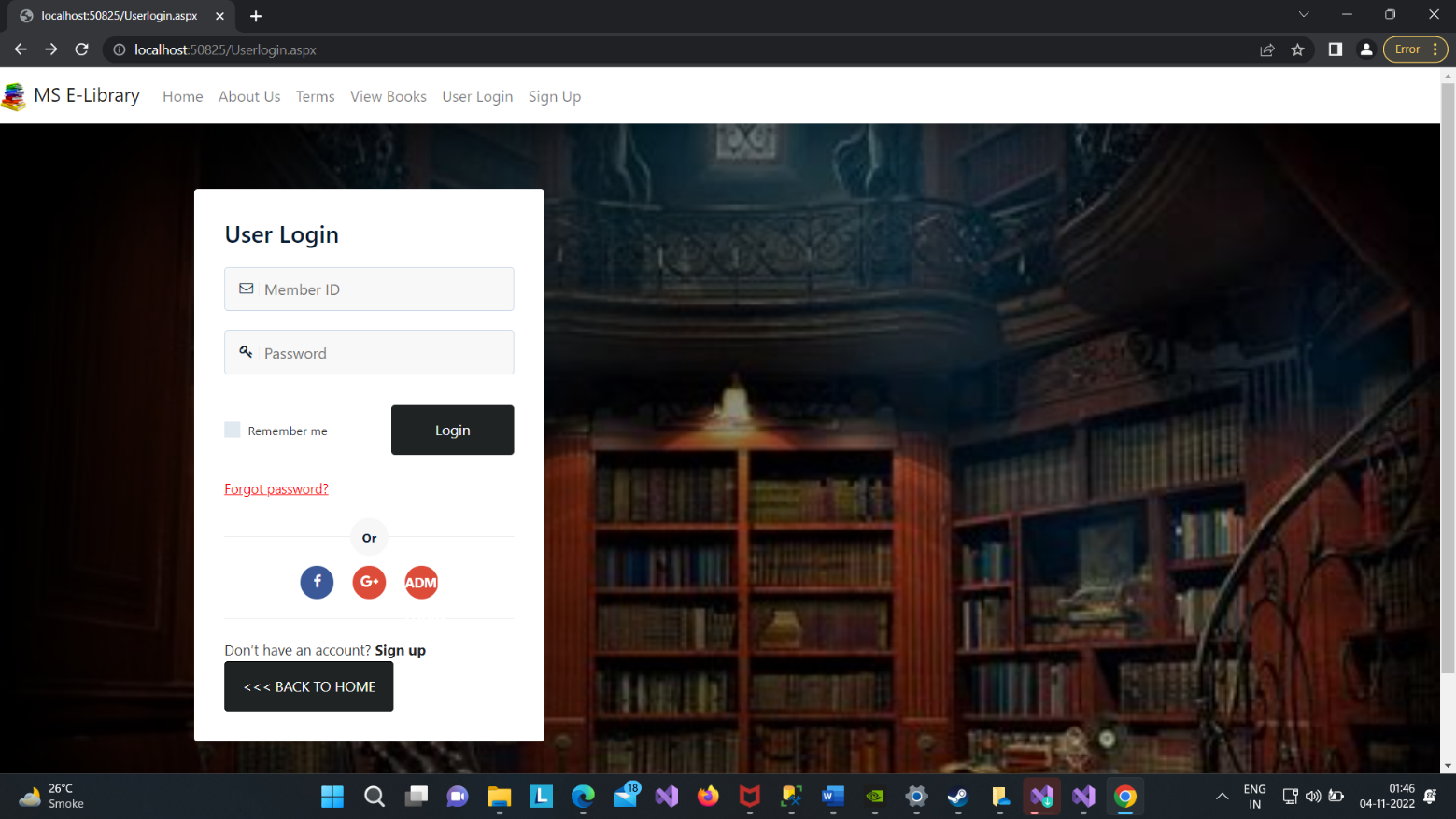
</div>

</div>

</asp:Content>

#### Code For Student login check

**4.1.2 Screenshot for Student login**



#### 4.1.3 Teacher Module

FORGOT PASSWORD

TEACHER LOGIN

TEACHER

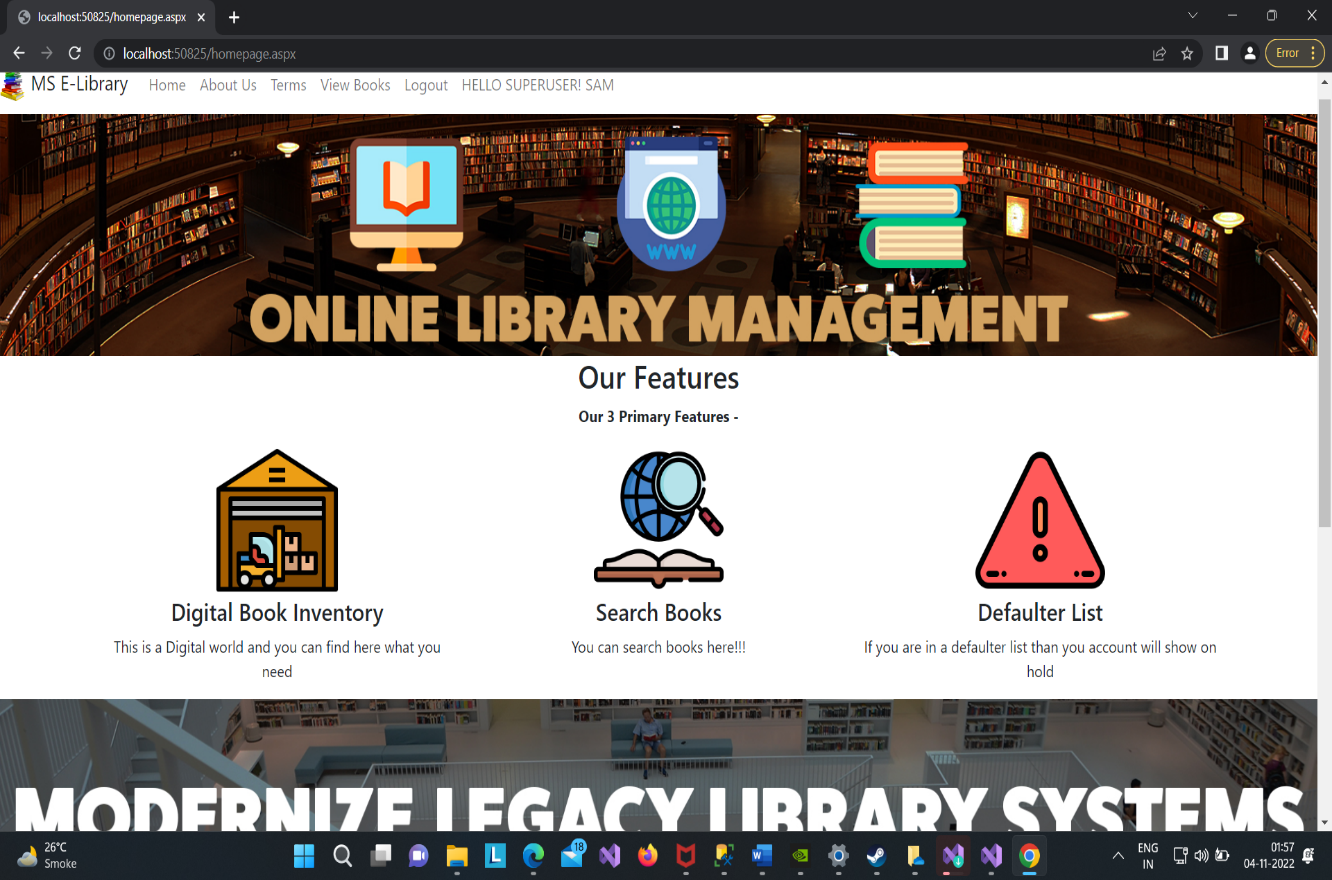
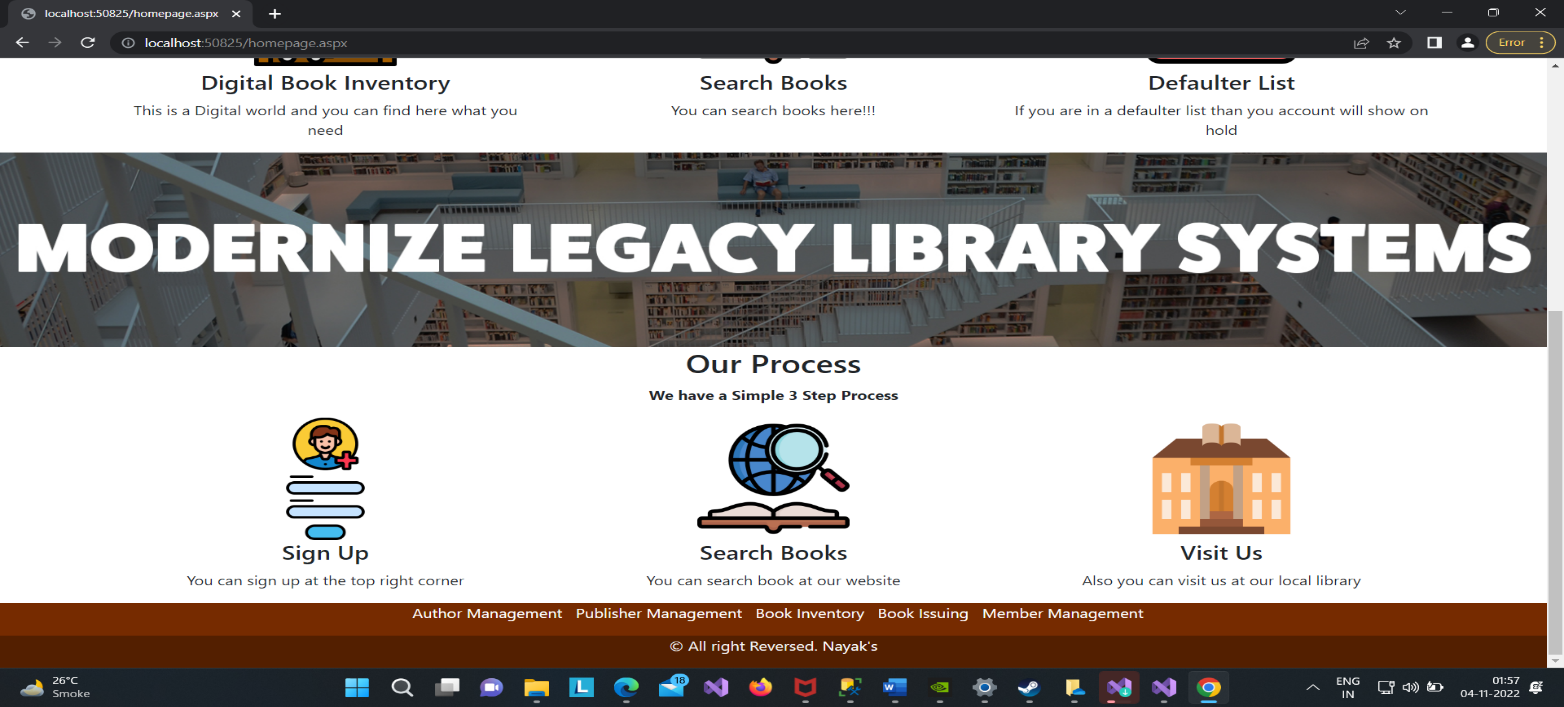
SUGGESTIONS

USERNAME AND PASSWORD AUTHENTICATION

EVENT ADDITION

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.

**4.1.4 Screenshot for Librarian login :-**

****

#### 4.1.4 Book Module

SEARCH BOOK

ADD NEW BOOK

BOOK MAINTENANCE

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

#### 4.1.5 Book Transaction Module

CHECK RESERVATION

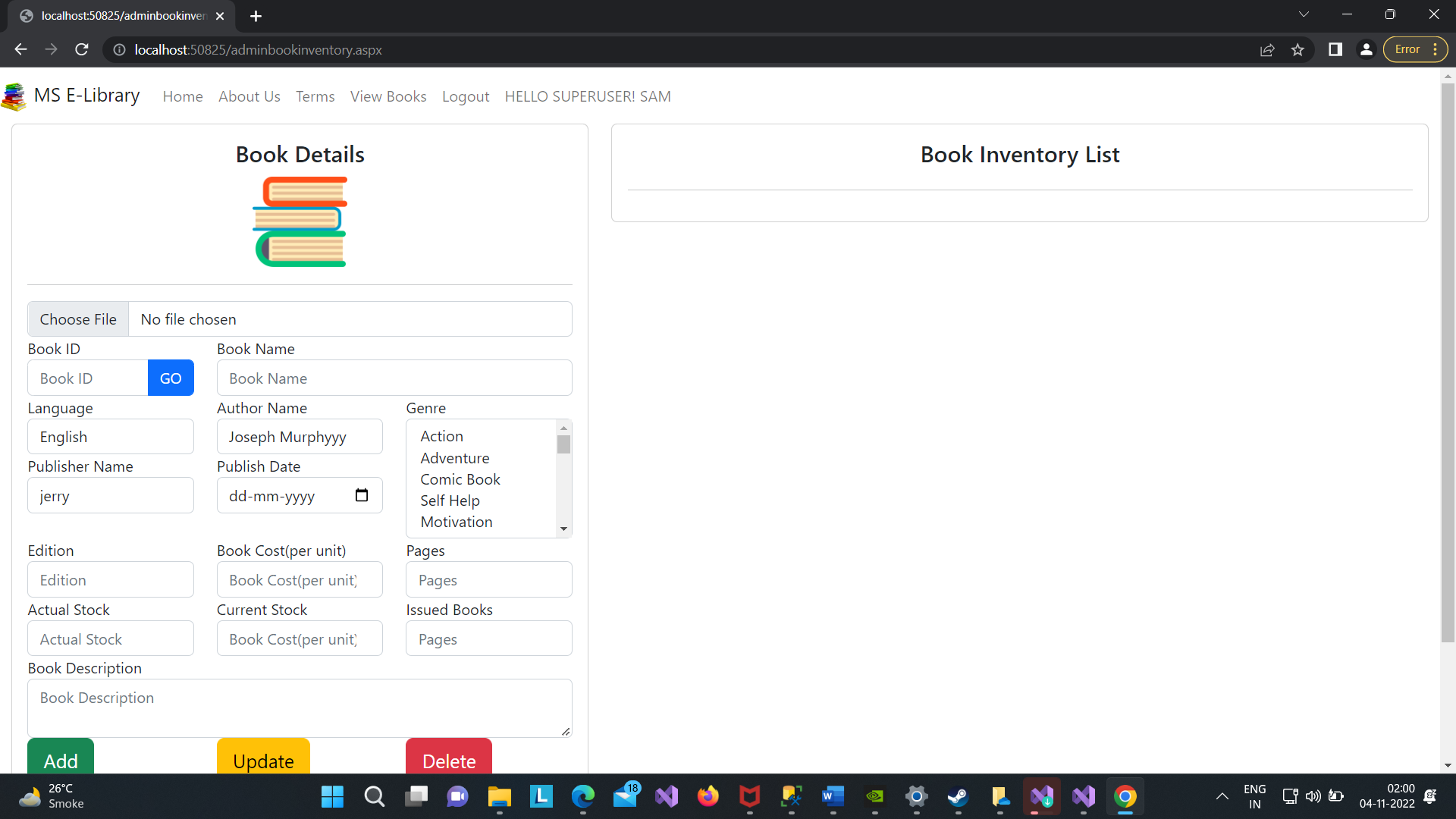
ISSUE BOOK

RETURN BOOK

BOOK TRANSACTION

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

**4.1.2 Screenshot for Book search**



#### 4.1.6 Report Module

BOOK REPORT

TRANSACTION REPORT

TEACHER REPORT

REPORT

STUDENT REPORT

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

# 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

1. 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 i.e besides the module we would require

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

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

1. Test For the admin module
   * Testing admin login form-This form is used for log in of administrator of the system.In this we enter the username and password if both are correct administration page will open other wise if any of data is wrong it will get redirected back to the login page and again ask for username and password
   * Student account addition- In this section the admin can verify student details from student academinc info and then only add student details to main library database it contains add and delete buttons if user click add button data will be added to student database and if he clicks delete button the student data will be deleted
   * 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 the library id, 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 library id, 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 log 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.bootstrap.com>
* <http://www.bootstrap.com>/theme
* <http://www.Gridview.com>
* <http://www.webfonts.com>