Index

Micro Project Proposal

LIBRARY MANAGEMENT SYSTEM

1. Aims/Benefits/Objectives of the Micro-Project:

The project aims and objectives that will be achieved after completion of this project are discussed as follows, the aims, objectives and benefits of the micro project are as follows:

- Online Book Issue

- Request column for librarian for providing new book

- A separate column for digital library

- Student login page where student can find books issued by him/her and date of return.

- A search column to search availability of books.

- A teacher login page where teacher can add any events being organized in the college and important suggestions regarding books.

- Online notice board about the workshop.

2. Course Outcome Addressed:

- CO-1: CO-2: Prepare the SRS, Design document, Project plan of a given software system
- CO-2: Solving the real-world problem using the software development models.
- CO-3: Explain process model for a software project development.

3. Proposed Methodology:

Library management system is an application which refers to library systems which are generally small or medium 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 a new books, addition of new students etc.

4. Action Plan:

Sr. No.	Details of Activity	Planned Start date	Planned Finish date	Name of Responsible Team Members
1	Search the topic	03-02-2023 2:00pm-3:00pm	07-02-2023 2:00pm-3:00pm	Akshay Dashrath Gitte & Harsh Moreshwar Kale & Kunal Nitin Nalwade
2	Search the information	10-02-2023 2:00pm-3:00pm	14-02-2023 2:00pm-3:00pm	
3	Project designing	21-02-2023 2:00pm-3:00pm	28-02-2023 2:00pm-3:00pm	
4	Project modeling	03-03-2023 2:00pm-3:00pm	07-03-2023 2:00pm-3:00pm	
5	Project developing	10-03-2023 2:00pm-3:00pm	14-03-2023 2:00pm-3:00pm	
6	Implementation	15-03-2023 2:00pm-3:00pm	17-03-2023 2:00pm-3:00pm	
7	Organizing the Project	18-03-2023 2:00pm-3:00pm	24-03-2023 2:00pm-3:00pm	
8	Finalizing Project with its report	28-03-2023 2:00pm-3:00pm	31-03-2023 2:00pm-3:00pm	

5. Resources Required:

Sr. No.	Name of resource / material	Specification	Quantity	Remarks
1	Computer	WINDOWS 11,8GB RAM	1	
2	Operating System	WINDOWS 11	1	
3	Software	Any UML Tool	1	
4	Browser	Google Chrome	1	

6. Names of Team Members with Roll No.'s:

Sr. No.	Enrollment No.	Name of Team Member	Roll No.
1	2110950049	Akshay Dashrath Gitte	01
2	2110950051	Harsh Moreshwar Kale	03
3	2210950151	Kunal Nitin Nalwade	49

Ms. Kachare S. M.

Name and Signature of the Teacher

Micro-Project Report

LIBRARY MANAGEMENT SYSTEM

1. Rationale:

The Library Management System is a software package that is useful to improve the efficiency of Libraries, Librarians and Users. The proposed product has interactions with various kinds of users -

1. Librarian

2. Members i.e. Students

3. Professors of ITM

2. Aims/Benefits of the Micro-Project:

- Library management systems are essentially relational databases customized for use in running a library and supporting its operations.
- Able to create an end-to-end software.
- Solving the real-world problem using the software development models.

3. Course Outcomes Achieved:

- CO-1: CO1- Select suitable Software Process model for software development.
- CO-2: Use Software modeling to create data designs.
- CO-3: Prepare software requirement specifications.

4. Literature Review:

INTRODUCTION

Library management system is an application which refers to library systems which are generally small or medium 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 a new books, addition of new students etc.

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

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

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

Requirements necessary for library management system project:

1) Processor - INTEL CORE PROCESSOR OR BETTER PERFORMANCE
2) Operating system – Windows 11
3) Memory – 8GB RAM, DDR4 and more.
4) Hard Disk Space - MINIMUM 8 GB FOR DATABASE USAGE FOR FUTURE
5) Data Base – MySQL

SYSTEM ANALYSIS

Here 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.

Software Requirement Specification:

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.

Some Problem Statements

1) File Lost
2) File Damaged due to system environmental problems
3) Difficulty in record searching
4) Space issues
5) Cost issues

SYSTEM REQUIREMENTS
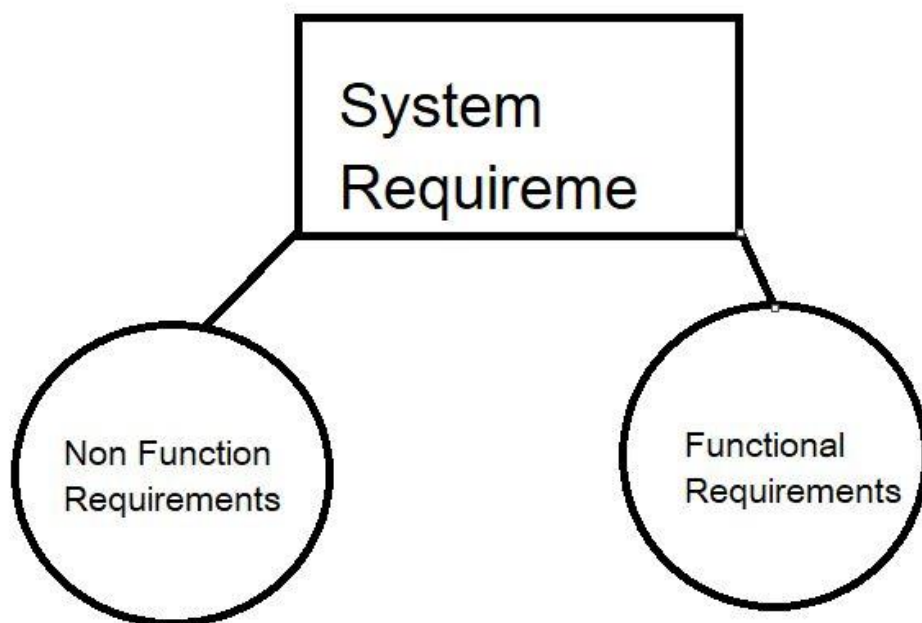
<u>Non-Functional Requirements:</u>

1) 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.

2) Reliability Requirement
   The system should accurately performs member registration, member validation,
   Report generation, book transaction and search.

3) 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.

4) Organizational Requirement
   i)   Implementation Requirements
        In implementing whole system it uses html in front end with php as server side
        Scripting language which will be used for database connectivity and the backend i.e.
        the database part is developed using mysql.
   ii) Delivery Requirements
        The whole system is expected to be delivered in six months of time with a weekly
        evaluation by the project guide.

<u>Functional Requirements:</u>

Normal Users

i)   User Login
     <u>Description of features</u>
     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.

     <u>Functional Requirements</u>
     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.

ii) Register New User
     <u>Description of features</u>
     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

iii)   Register New Book
Description of features
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.

iv)   Search Book
Description of features
This feature is found in book maintenance part. We can search book based on book id, book and publication or by author name.

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

v)   Issue Books & Return Books
Description of features
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

vi)   Event Addition
Description of features
This feature allows teacher and student to add information about various workshops being conducted in college and colleges nearby.

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

Software Tool Used

     In this project we need HTML, CSS & JS for front end. By using the hypertext markup language we can able to markup the require skeleton or structure for the application. When this markup language is used we need to present it in a user friendly manner so we need CSS for making it more beautiful in sense of presentation. Finally we used JS to add events and control on the page or in application.

If we are thing about backend as the client server structure is responsible for the serving the files to the user so we can use PHP (Hypertext Preprocessor) to hide the internal implementation and showing the essential output to the user we can use PHP and database likely MONGODB, MYSQL!

SYSTEM DESIGN

BOOK TABLE FOR KEEPING TRACK OF BOOKS

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

STUDENT TABLE FOR STUDENT INFORMATION

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

TEACHER TABLE TO KEEP TEACHER INFORMATION

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

ISSUE TABLE TO KEEP TRACK OF BOOKS ISSUED

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

STUDENT LOGIN TABLE

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

EVENT TABLE FOR EVENT INFORMATION

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

TEACHER LOGIN TABLE

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

Data flow diagrams for Library Management System

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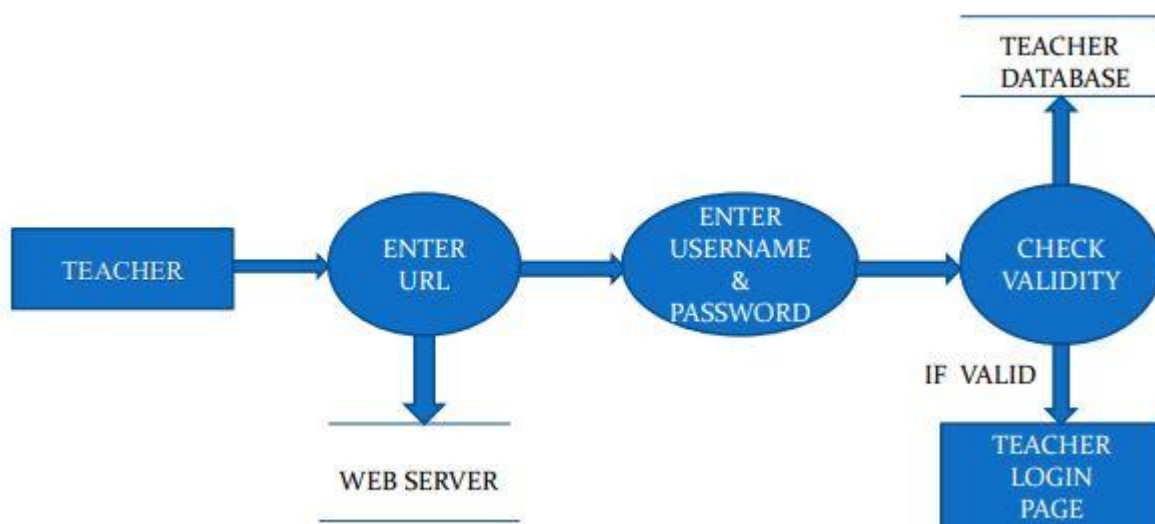
After entering to the home page of the website, teacher can choose the TEACHER LOGIN Option where they are asked to enter username & password, and if he/she is a valid user then a teacher login page will be displayed.
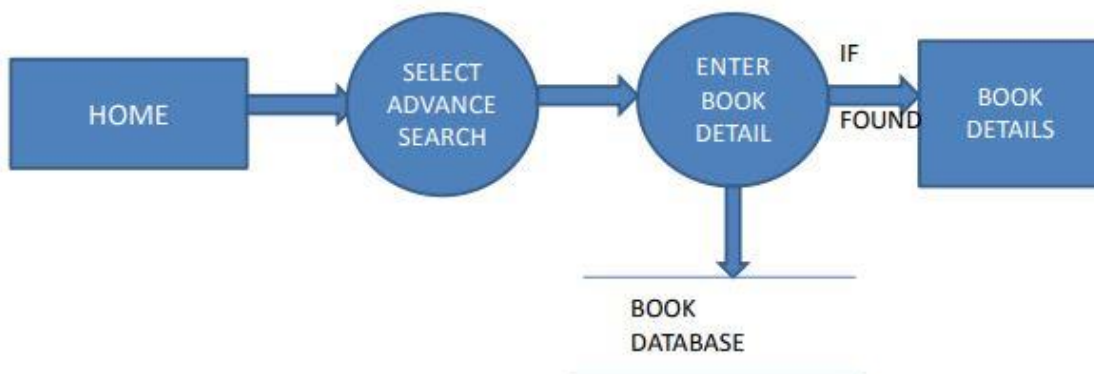
DATA FLOW DIAGRAM FOR STUDENT LOGIN:
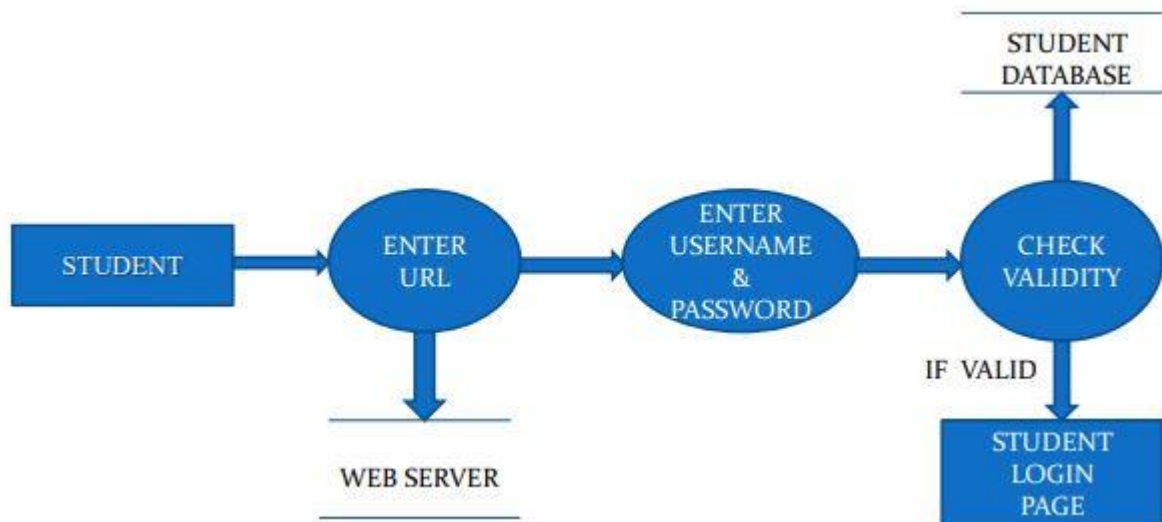
After entering to the home page of the website, student can choose the STUDENT LOGIN Option where they are asked to enter username & password, and if he/she is a valid user then a student login page will be displayed.

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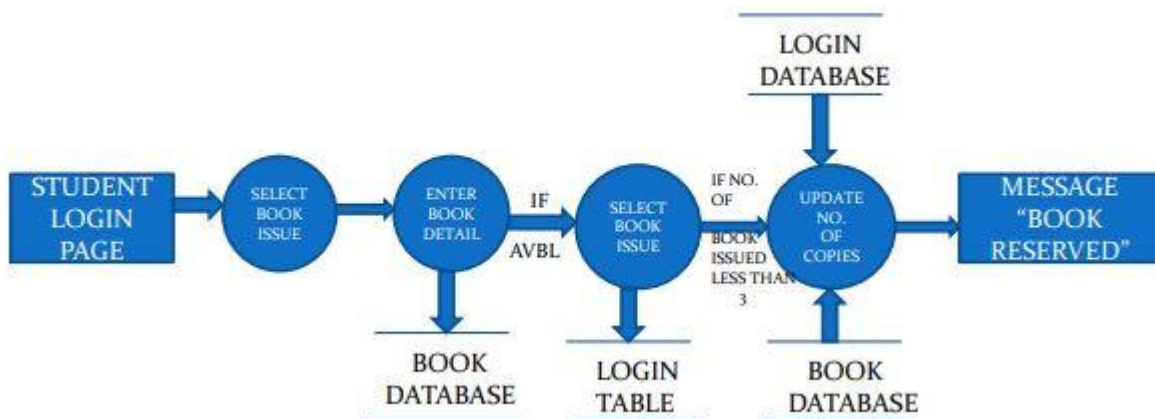
It is a 2nd level Data Flow Diagram where after entering STUDENT LOGIN page he/she can select a book issue option where after entering the book detail, he/she can select the book issue option and if the maximum no of books issued limit is not crossed then a request will be sent to the librarian who will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

DATA FLOW DIAGRAM FOR BOOK SEARCH:

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

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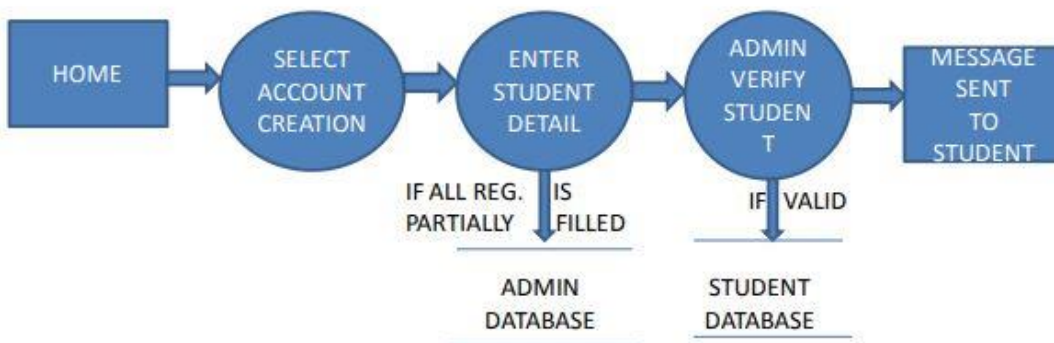
After the home page login there will be an option of CREATE AN ACCOUNT where after entering student detail, if all the fields are filled then a request will be sent to the librarian who will approve him as a registered member of the library.

SYSTEM IMPLEMENTATION

Screenshot of the Home Page:

Module Description

For Library Management System it is divided into the Following Basic Two Modules:

Code for Home Page Module:

```html
index.html > ...
1      <!doctype html>
2      <html lang="en">
3
4      <head>
5          <meta charset="utf-8">
6          <meta name="viewport" content="width=device-width, initial-scale=1">
7          <meta name="Manager" content="Akshay Dashrath Gitte">
8          <meta name="Developer" content="Harsh Moreshwar Kale">
9          <meta name="Tester" content="Kunal Nittin Nalwade">
10         <title>Library Management System!</title>
11         <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha3/dist/css/bootstrap.min.css"
           rel="stylesheet"
12             integrity="sha384-KK94CHFLLe+nY2dmCWGMq91rCGa5gtU4mk92HdvYe+M/SXH301p5ILy+dN9+nJOZ"
               crossorigin="anonymous">
13         <link rel="shortcut icon" href="favicon.jfif" type="image/x-icon">
14         <link rel="stylesheet" href="style.css">
15     </head>
16
17     <body>
18         <!-- Navbar -->
19         <nav class="navbar navbar-expand-lg bg-body-tertiary">
20             <div class="container-fluid">
21                 <a class="navbar-brand" href="index.html">Library Management System!</a>
22                 <button class="navbar-toggler" type="button" data-bs-toggle="collapse"
23                     data-bs-target="#navbarSupportedContent" aria-controls="navbarSupportedContent"
                       aria-expanded="false"
```

```html
                    aria-label="Toggle navigation">
                    <span class="navbar-toggler-icon"></span>
                </button>
                <div class="collapse navbar-collapse" id="navbarSupportedContent">
                    <ul class="navbar-nav me-auto mb-2 mb-lg-0">
                        <li class="nav-item">
                            <a class="nav-link active" aria-current="page" href="index.html">Home</a>
                        </li>
                        <li class="nav-item">
                            <a class="nav-link" href="books.html">Books</a>
                        </li>

                    </ul>
                    <form class="d-flex" role="search">
                        <input class="form-control me-2" type="search" placeholder="Search"
                        aria-label="Search">
                        <button class="btn btn-outline-success" type="submit">Search</button>
                    </form>
                </div>
            </div>
        </nav>

        <!-- Message Div -->
        <div id="message"></div>

```
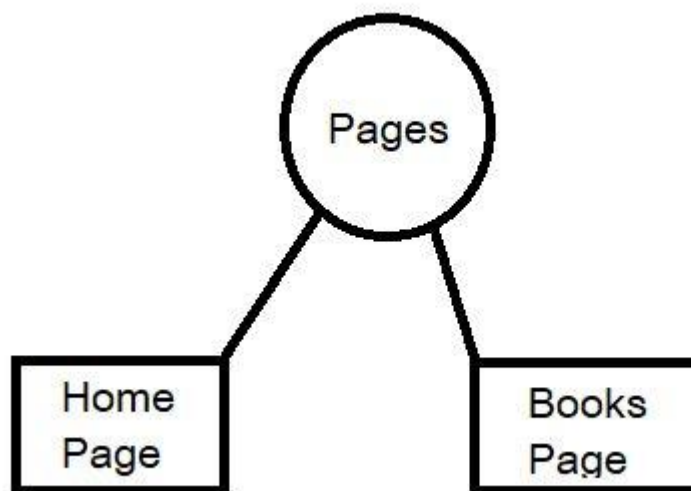```html
        <!-- Main -->
        <div class="container">
            <h2><b>Library Management System!</b></h2>
            <hr>
            <form id="libraryForm">
                <div class="form-group row">
                    <label for="bookName" class="col-sm-2 col-form-label">Name</label>
                    <div class="col-sm-10">
                        <input type="text" class="form-control" id="bookName" placeholder="Book
                        Name">
                    </div>
                </div>
                <div class="form-group row">
                    <label for="Author" class="col-sm-2 col-form-label">Author</label>
                    <div class="col-sm-10">
                        <input type="text" class="form-control" id="author" placeholder="Author">
                    </div>
                </div>
                <fieldset class="form-group">
                    <div class="row">
                        <legend class="col-form-label col-sm-2 pt-0">Type</legend>
                        <div class="col-sm-10">
                            <div class="form-check">
                                <input class="form-check-input" type="radio" name="type" id="SEN"
                                value="Software Engineering"
```

```html
                                    checked>
                                <label class="form-check-label" for="SEN">
                                    Software Engineering
                                </label>
                            </div>
                            <div class="form-check">
                                <input class="form-check-input" type="radio" name="type"
                                    id="programming"
                                    value="Computer Programming">
                                <label class="form-check-label" for="programming">
                                    Computer Programming
                                </label>
                            </div>
                            <div class="form-check disabled">
                                <input class="form-check-input" type="radio" name="type" id="EC"
                                    value="Electronics & Computers">
                                <label class="form-check-label" for="EC">
                                    Electronics & Computers
                                </label>
                            </div>
                        </div>
                    </div>
                </fieldset>

                <div class="form-group row">
                    <div class="col-sm-10">
                        <button type="submit" class="btn btn-primary">Add Book</button>
                    </div>
                </div>
            </form>

            <br>

            <div id="table">
                <h1>Your Books</h1>
                <table class="table">
                    <thead>
                        <tr>
                            <th scope="col">Name</th>
                            <th scope="col">Author</th>
                            <th scope="col">Type</th>
                        </tr>
                    </thead>
                    <tbody id="tableBody">

                    </tbody>
                </table>
            </div>
        </div>
```

16

```
118
119        <div class="footer">
120            <footer>
121                <p>Library Management System &copy; 2023</p>
122            </footer>
123        </div>
124
125        <script src="script.js"></script>
126
127        <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha3/dist/js/bootstrap.bundle.
           min.js"
128            integrity="sha384-ENjdO4Dr2bkBIFxQpeoTz1HIcje39Wm4jDKdf19U8gI4ddQ3GYNS7NTKfAdVQSZe"
129            crossorigin="anonymous"></script>
130        <script src="https://cdn.jsdelivr.net/npm/@popperjs/core@2.11.7/dist/umd/popper.min.js"
131            integrity="sha384-zYPOMqeu1DAVkHiLqWBUTcbYfZ8osu1Nd6Z89ify25QV9guujx43ITvfi12/QExE"
132            crossorigin="anonymous"></script>
133        <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha3/dist/js/bootstrap.min.js"
134            integrity="sha384-Y4oOpwW3duJdCWv5ly8SCFYWqFDsfob/3GkgExXKV4idmbt98QcxXYs9UoXAB7BZ"
135            crossorigin="anonymous"></script>
136    </body>
137
138  </html>
```

Code for Adding Logic in Website:

```
JS script.js > ⦿ libraryFormSubmit
 1     console.log("This is Script.js!");
 2
 3     // Constuctor
 4     function Book(name, author, type) {
 5         this.name = name;
 6         this.author = author;
 7         this.type = type;
 8     }
 9
10     // Display Constructor
11     function Display() {
12
13     }
14
15     // Add methods to display prototype.
16     Display.prototype.add = function (book) {
17         console.log("Adding to GUI!!");
18         tableBody = document.getElementById("tableBody");
19         let uiString = `
20             <tr>
21                 <td>${book.name}</td>
22                 <td>${book.author}</td>
23                 <td>${book.type}</td>
24             </tr>
25         `;
26         tableBody.innerHTML += uiString;
```

```
52          message.innerHTML = '';
53      }, 2000);
54  }
55
56
57  // Add submit event listener to libraryForm.
58  let libraryForm = document.getElementById("libraryForm");
59  libraryForm.addEventListener("submit", libraryFormSubmit);
60
61  function libraryFormSubmit(e) {
62      e.preventDefault();
63      console.log("You have submitted library form!");
64      let name = document.getElementById("bookName").value;
65      let author = document.getElementById("author").value;
66      // [SEN, programming, EC].forEach(Element=>document.getElementById(Element));
67      let type;
68      let SEN = document.getElementById("SEN");
69      let programming = document.getElementById("programming");
70      let EC = document.getElementById("EC");
71
72      if (SEN.checked) {
73          type = SEN.value;
74      }
75      else if (programming.checked) {
76          type = programming.value;
77      }
```

```
78      else if (EC.checked) {
79          type = EC.value;
80      }
81
82
83      let book = new Book(name, author, type);
84      console.log(book);
85
86      let display = new Display();
87      if(display.validate(book)){
88          // localStorage.setItem("book", display.add(book));
89          display.add(book);
90          display.clear();
91          display.show("success", " Your Books is Succesfully Added!")
92      }else{
93          display.show("danger", " Sorry, You cannot add this Book!")
94      }
95  }
```

Code for Styling the Pages of website:

```
style.css > ...
1   *{
2       margin: 0;
3       padding: 0;
4       box-sizing: border-box;
5   }
6
7   .container{
8       overflow-y: scroll;
9       height: 100vh;
10  }
11
12  .footer{
13      position: relative;
14      margin-top: 20px;
15      height: 40px;
16      background: #f8f9fa;
17      text-align: center;
18  }
```

SYSTEM TESTING

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

Our Project went through two levels of testing.

1) Unit Testing
2) Integration Testing

Unit Testing

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

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

Here, in the unit testing we test each and every module in this case our module is a page so iterate through each page by testing it by debugging the code.

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.

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.

Output :

5. Actual Resources Used:

Sr. No.	Name of resource / material	Specification	Quantity	Remarks
1	Computer	WINDOWS 11,8 GB RAM	1	
2	Operating System	WINDOWS 11	1	
3	Software	Any UML Tool	1	
4	Browser	Google Chrome	1	

6. Skill developed / Learning out of this Micro-Project:

- Able to create a full flexible Graphical User Interface Application.
- Able to follow or complete the step by step procedure of software development process

  by satisfying all needs for solving a real-world problem.

7.Applications of this Micro-Project:

By using the Software Development Model we are able to build a full stack website for arrange and storing, retrieving the books. To able to push this library management system into user environment with wide functions that are helpful to each and every user.

8. References:

Library Management System! (library-management-system-vapm.netlify.app)

Harshk133 (Harsh Moreshwar Kale) (github.com)

http://www.w3schools.com/html/html_intro.asp

http://www.w3schools.com/css/css_background.asp

http://www.w3schools.com/js/js_datatypes.asp

http://www.w3schools.com/sql/sql_insert.asp

http://www.w3schools.com/sql/sql_update.asp

http://www.w3schools.com/php/php_forms.asp

Fundamentals of software engineering by Rajib mall, PH learning (this is for backend but not included in our project for some technical reasons)

Web development and application development by Ivan Byross BPB publications