Texas International College

Tribhuvan University

Faculty of Humanities and Social Science

LIBRARY MANAGEMENT SYSTEM

A PROJECT REPORT

Submitted to

Department of Computer Application

In partial fulfillment of the requirements for the Bachelor of Computer Application

Submitted By

Rojal Luitel(6-2-926-21-2020),

Rajak Putuwar(6-2-926-17-2020)

BCA 4th, 2077 Batch

SUPERVISOR'S RECOMMENDATION

I hereby recommend the project "**LIBRARY MANAGEMENT SYSTEM**" for final evaluation. This collaborative endeavor developed under my guidance, showcases the dedication of **Rajak Putuwar (6-2-926-17-2020)** and **Rojal Luitel (6-2-926-21-2020)**.

I have no doubt that their efforts and the quality of this project deserve recognition. Their diligence and perseverance have been evident at every stage, from initial concept to final implementation.

Therefore, I confidently recommend this project for evaluation.

……..

Mr. Omkar Basnet

Supervisor

Texas Int'l College

Chabahil, Kathmandu

LETTER OF APPROVAL

We hereby affirm that the project, jointly developed by **Rajak Putuwar (6-2-926-17-2020)** and **Rojal Luitel (6-2-926-21-2020)**, titled **"LIBRARY MANAGEMENT SYSTEM"** and submitted in partial fulfillment of the requirements for the Bachelor of Computer Application, has been thoroughly reviewed. In our assessment, we find this project to be adequate in its breadth and quality to meet the requirements for the awarded degree.

…….

Mr. Omkar Basnet

Supervisor

Texas Int'l College

Chabahil, Kathmandu

…….

Mr. Omkar Basnet

HOD, BCA

Texas Int'l College

Chabahil, Kathmandu

…….

Mr. Sulav Nepal

Internal Examiner

…….

External Examiner

ABSTRACT

With the advancement of technology, it is imperative to exalt all the systems into a user-friendly manner. The library management System acts as a tool to transform traditional libraries into digital libraries. In traditional libraries, the students/user has to search for books which are hassle process and there is no proper maintenance of database about issues. The overall progress of work is slow and it is impossible to generate a fast report.

The librarian has to work allotted for arranging, sorting books in the book sells. At the same time, they have to check and monitor the lend/borrow book details with its fine. It is a tedious process to work simultaneously in different sectors.

Library management system will assist the librarians to work easily. The library Management System supports the librarians to encounter all the issues concurrently. The users need not stand in a queue for a long period to return/borrow book from the library. The single pc contains all the data in it. The librarians have to assess the system and provide an entry in it.

Through LMS the librarian can find the book in the bookshelves. The LMS is designed with the basic features such as librarian can add/view/update/delete books and students' details in it. Once he/she ingress into the system they can modify any data in the database.

ACKNOWLEDGEMENT

We want to express our heartfelt thanks to several individuals who played crucial roles in the successful completion of our "LIBRARY MANAGEMENT SYSTEM" project. We our warm thanks to our esteemed supervisor, Mr. Omkar Basnet, for his invaluable guidance and advice during important moments, guiding us in the right direction for project completion.

Furthermore, we are immensely thankful to our HOD, Mr. Omkar Basnet and our faculty members Mr. Sulav Nepal, and Mr., Kumar Poudyal for their continuous support throughout the project. We also extend our sincere appreciation to our senior colleagues and friends for their unwavering guidance and support during this project's journey.

While we devoted significant efforts to this project, it would not have been possible the generous support of numerous individuals and organizations. Our gratitude extends to all who contributed. We are deeply indebted to Texas International College for their guidance, ongoing supervision, provision of essential project-related information and resources, and support in completing this project. We also express our thanks to our family members, college staff, and faculty for their cooperation and encouragement, which boosted our project's completion.

Sincerely,

Rajak Putuwar (6-2-926-17-2020)

Rojal Luitel (6-2-926-21-2020).

TABLE OF CONTENT

LIST OF ABBREVIATIONS

LMS : Library Management System

SQL : Structured Query Language

HTML : Hyper Text Markup Language

DB : Database

CSS : Cascading Style Sheet

DFD : Data Flow Diagram

ER : Entity Relation

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1: INTRODUCTION

1.1. Introduction

A library management system is a web-based application designed to maintain, organize, and handle countless books systematically. It can track the numbers of books in the library, issued books, and returned books. If the due date is past, it has the function of charging the fine. The users can find the books in an instant and even reserve them through this system. This will help eliminate repetitive manual work and minimize the chance of errors and the immense use of paperwork. Earlier, the librarian used to manage the whole work in manual mode in the form of files and record books. Also, the process of adding new books, new users, issuing, and returning books had to be managed in a manual manner, which is very slow and inefficient. This application resolves this problem and provides a better solution. Users can access the system from any location with an internet connection to search for the required books. This web application can track the books taken by the user and their due date for return. And the users will also get notifications about the new arrival of books and about returning books.

1.2. Problem statement

In case of manual search of books, it experiences high cost of time loss for both librarian and user. Keeping of manual record may cause loss of record in case of any physical or natural harm. Managing records of incoming and outgoing of books manually is very tedious. Users get bad experience visiting library in search of same book repeatedly. Users have to loss their precious time just to know that the library doesn’t have a book that the user is seeking for.

1.3. Objectives

Due to the problem in the manual library management, we propose this system with the following objectives:

- To reduce the time required to maintain library functions manually by making it computerized.
- To search the books easily with easy update and adding in database.
- To calculate the fine and notify the user regarding fine through mails.

1.4. Scope and Limitations

The "Library Management System" project aims to revolutionize the way user's access and navigate their learning environment. It encompasses the following key areas:

- Provide the list of books the users can borrow. Users do not have to wait for a long to the book they want to borrow the system inbuild search function get the job done no time.
- Facility to reserve books that are available. Users do not have to visit library manually and can easily reserve book that are available in library.
- A status page for all users to view books borrowed by them, their individual due dates and their individual penalties if any.
- Providing interface to add or delete books: The system allow admin to add or delete the books according to the requirement of user.

Limitations:

- Coverage of Educational Institution: While the project will aim to include a wide range of colleges and universities, it may not cover every educational institution outside Kathmandu valley.
- Real-Time Updates: Users may not be updated by librarian or admin regarding the recommendation of books in library.

CHAPTER 2: BACKGROUND STUDY AND LITERATURE REVIEW

2.1. Background Study

Throughout history, library has evolved from manually managing to the integration of technology. The project acknowledges the impact of this evolution on how librarian or admin can manage book and user's access the books. In the present situation, librarian as well as users have to face different difficulties in managing library and accessing the books. The project recognizes the need for simplifying managing and accessing of books.

A library management system typically comprises several key components that work together to facilitate the management of library resources, services, and operations. Sure, a library management system is like the brain of a library. It has a big database that holds all the important info like books, who's borrowing them, and what's available. There's a special interface for people who work there and for folks who visit. It lets you search for books, see if they're in, reserve them, and handle your account. Librarians use a special tool to organize everything neatly details about books, who wrote them, and where they belong. Also, there's a part that keeps track of who's borrowing what, when it's due back, any fines, and the rules about borrowing stuff. All these parts work together smoothly to make the library run really well.

2.2. Literature review

Most of the research and writings on library management have focused on academic libraries and only recently has there been more interest in the administration of public libraries. The skill and style of public library managers, the directors, branch managers, and department and service managers who are leading these institutions – strongly affects the culture of a public library [1].

Library staff looks to these managers to help them navigate through the rapid changes that are occurring in public libraries as these changes in technology, roles, and user expectations strongly alter their daily routines of public service. Contemporary library managers need a wider array of skills and attributes than their earlier and more traditional counterparts and will need to seek continual professional development to remain effective as public libraries transition into the twenty-first century [2].

These managers will also need to distinguish between management and leadership skills and learn to identify and mentor leaders within their staff who can assist in the transition. This paper is a brief scan of the literature currently available on managing libraries and includes information on academic as well as public libraries due to the above-mentioned lack of public library material.

Before we build our project, we first analyze other existing systems.

2.2.1. Blackboard Library Management:

It is one of the library management systems used in Nepal. It has the most of the exiting features. dashboard that displays a wide range of information about the operations of the school library. The system organizes books by the level of reading, determines the reading level of each book, determines the genre for each book. The library management system allows students to borrow books from the library when they need them. System has a history with books in the library starting with writing development and other inventions like paper and book printing.

CHAPTER 3: SYSTEM ANALYSIS AND DESIGN

3.1 System Analysis

3.1.1 Requirement Analysis

I. Functional Requirements

Table 1: List of Functional Requirements

Requirement Id	Functional Requirement	Description
FR1	Add Book	Allow admin to add books,
FR2	Book Search	Allow admin as well as user to search book,
FR3	Book Booking	Allow user to book books,
FR4	Book Category	Allow admin to add category of books,
FR5	Book Issue	Allow admin to issue book to users,
FR6	Login	Allow users as well as admin to enter into system if they are authenticated,
FR7	Sign Up	Allow user to register into system
FR8	Book List	Allow admin and users to view all the books in library.

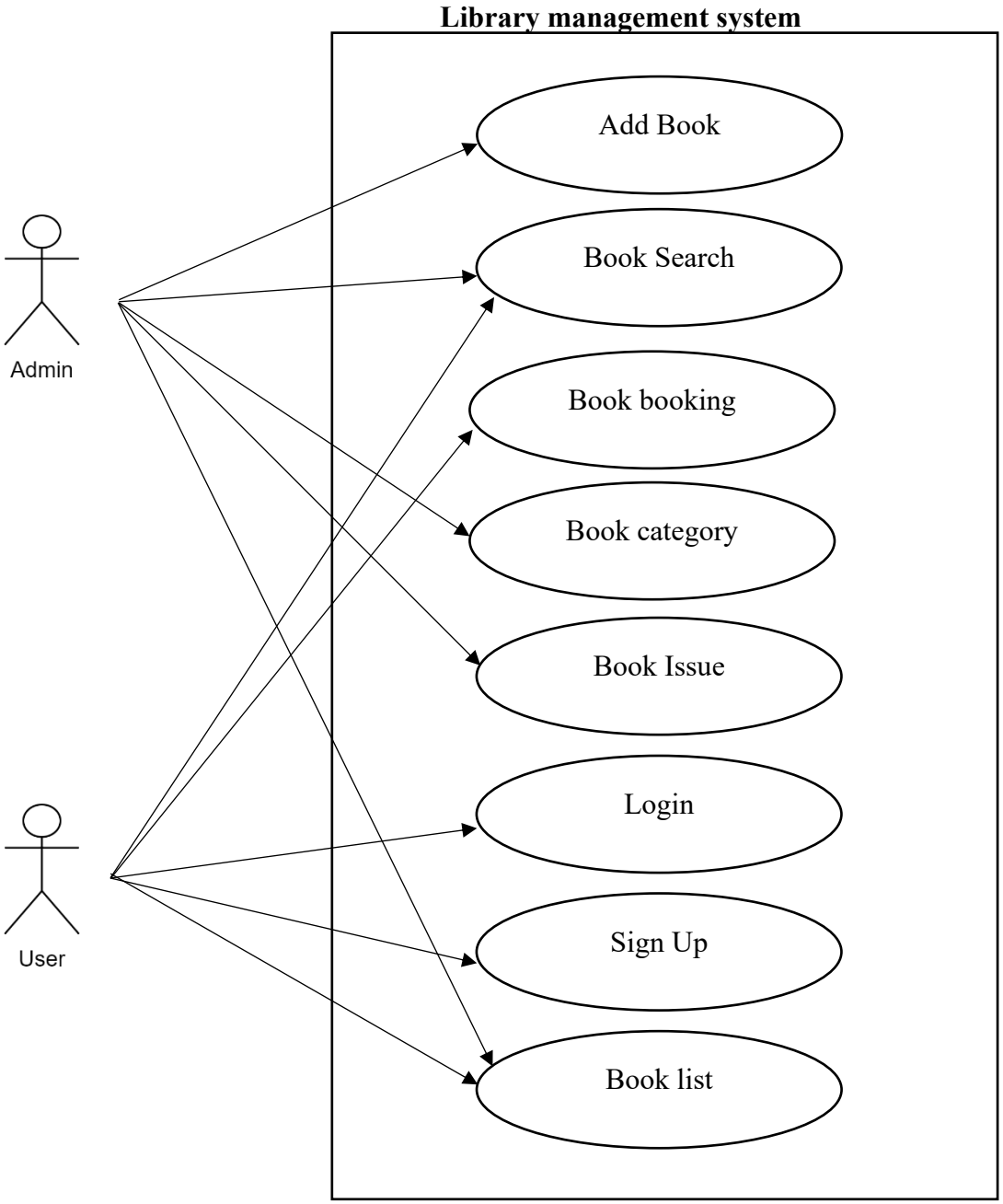


Fig 3.1.: Use Case Diagram of LMS

II. Non-Functional Requirement

Following are the non-functional requirement of library management system:

- **Performance**: This project is for the specific college or school which does not contain large amount of data. So, our system is fast.
- **Security:** In order to get access into the system, Users should first login and then only they are allowed to operate.

- **User-friendly**: Our design is optimal and optimized so, end user can easily understand and use the interface.
- **Scalability**: We can later on expand the project and make it online. We use Laravel framework so that we can add module and expand further**.**

3.1.2 Feasibility test

Technical:

HTML/CSS is used for frontend and Laravel is user for backend, MySQL for database.

Hardware to operate like laptop and mobile with web browser installed.

Operational:

Our project can solve basic operation performed in library like adding book, issuing books, listing books, check availability of books. So, it is operationally feasible.

Economic:

We use our own resources like (net and laptop) to propose the project. So, it is economically feasible.

Schedule:

Our system was proposed within the time frame.

3.1.3 Data Modeling

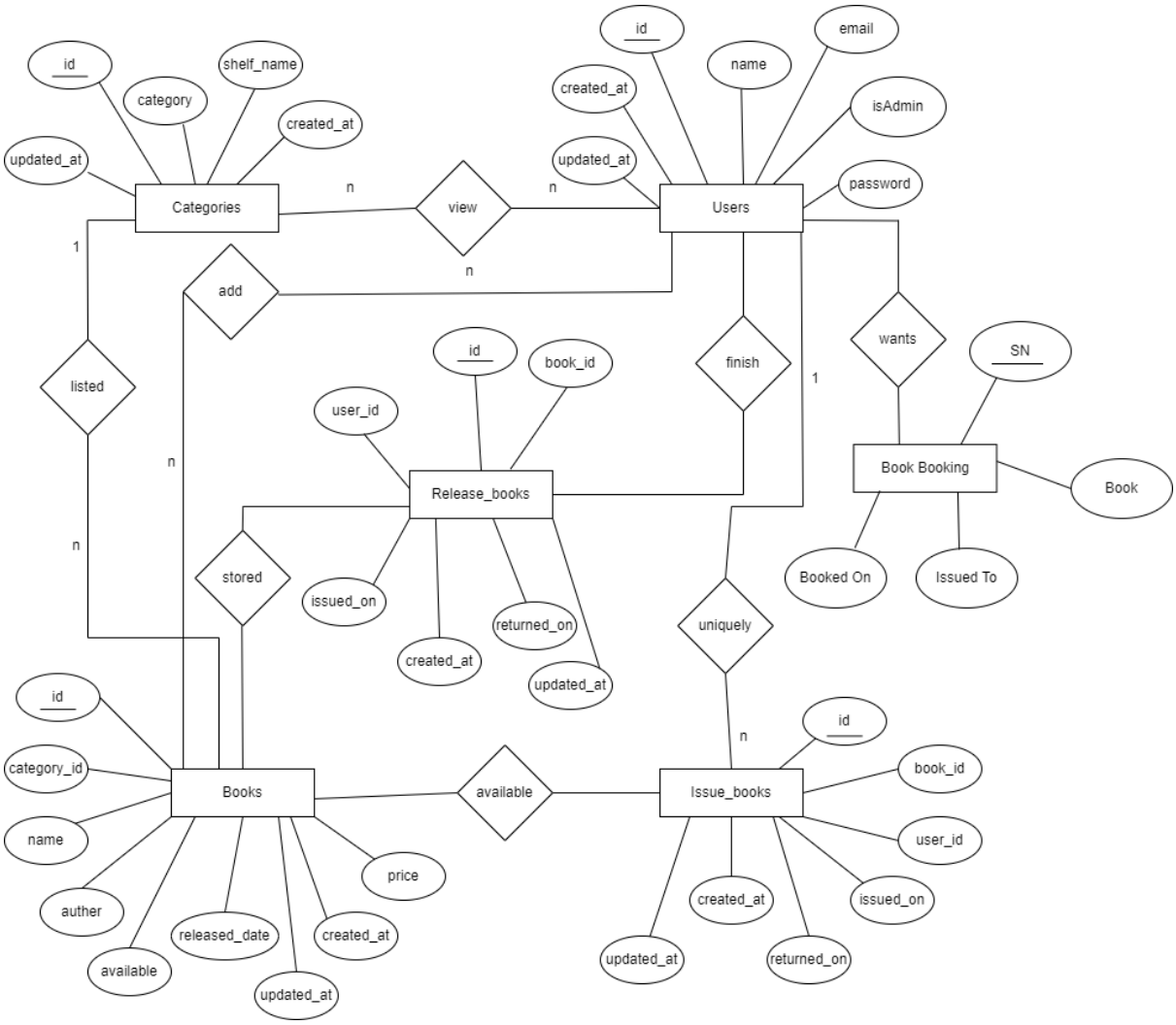


Fig3.2.: ER Diagram of LMS

3.1.4 Process modeling

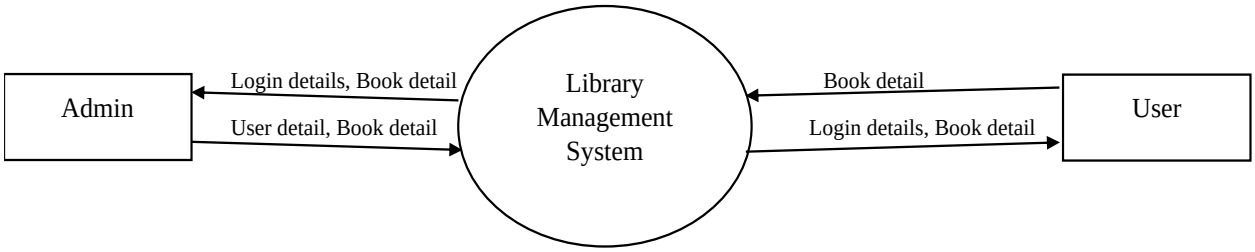


Fig3.3.: Context level Diagram for Library management system

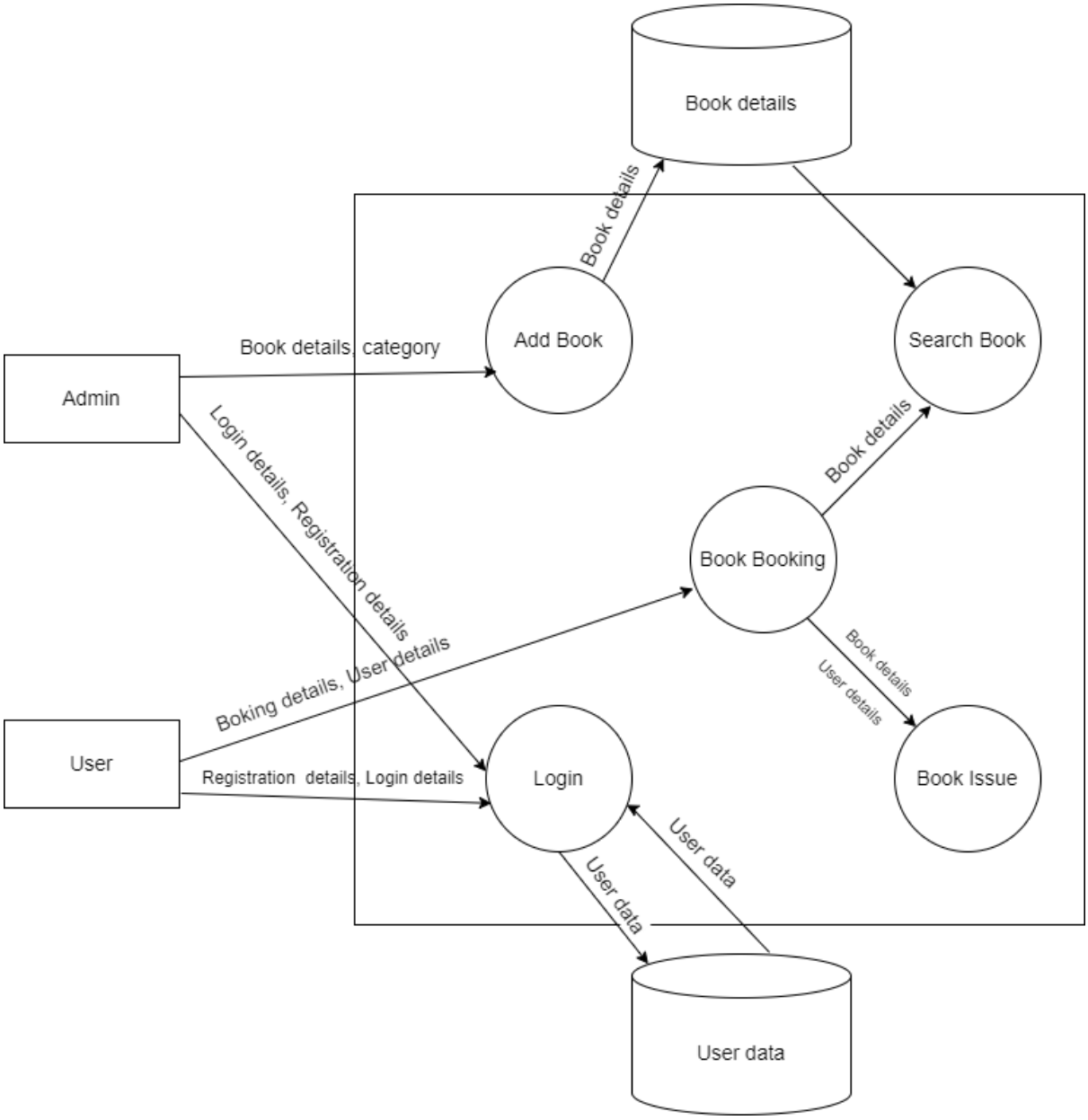


Fig 3.4. 0 level DFD of LMS

3.2. System Design

Creating a library system is like building a puzzle with lots of pieces. We carefully plan how everything fits together like where to store information, how people will use the system, and making sure different parts work well together. We think about making it grow without breaking, keeping it safe, and running smoothly. We test it a lot and have plans to fix things if they go wrong. To make it easy for both the librarians and the users finding what they need.

3.2.1. Architectural design

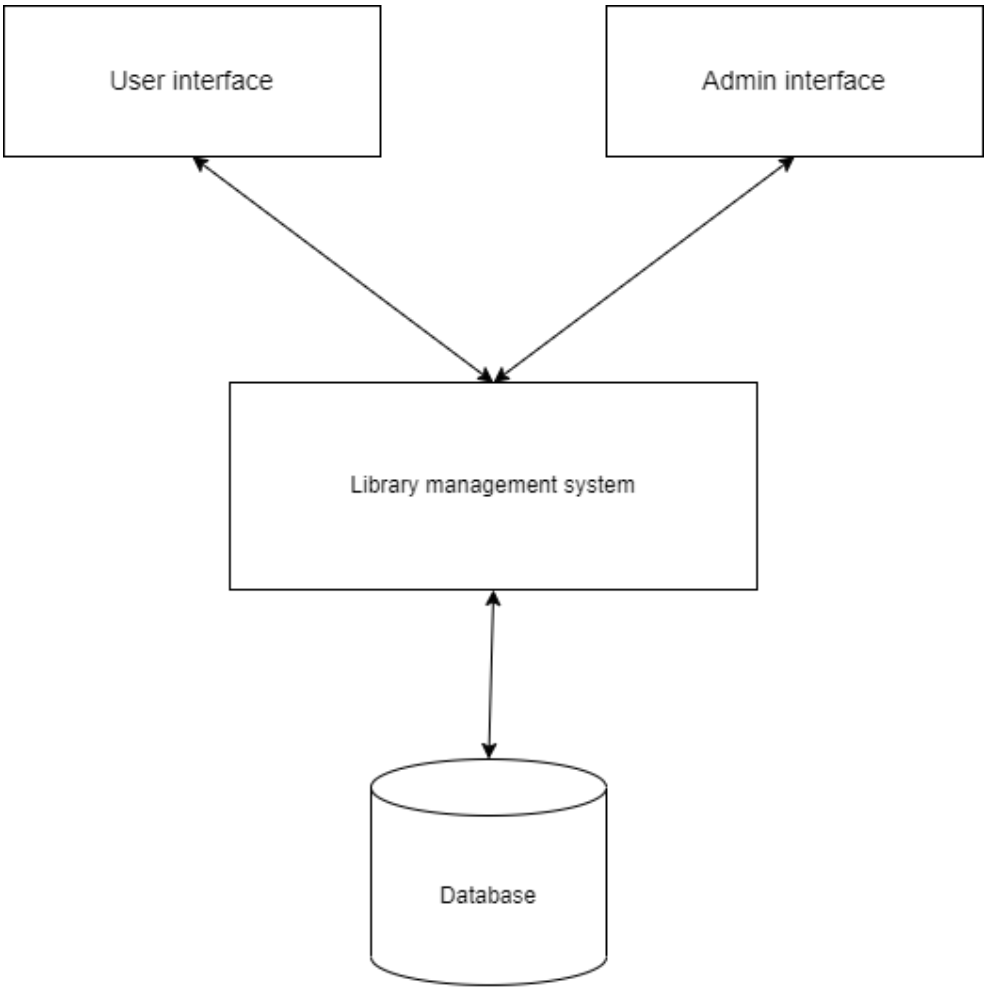


Fig 3.5.: Architectural Design of LMS

3.2.2 Database Schema Design

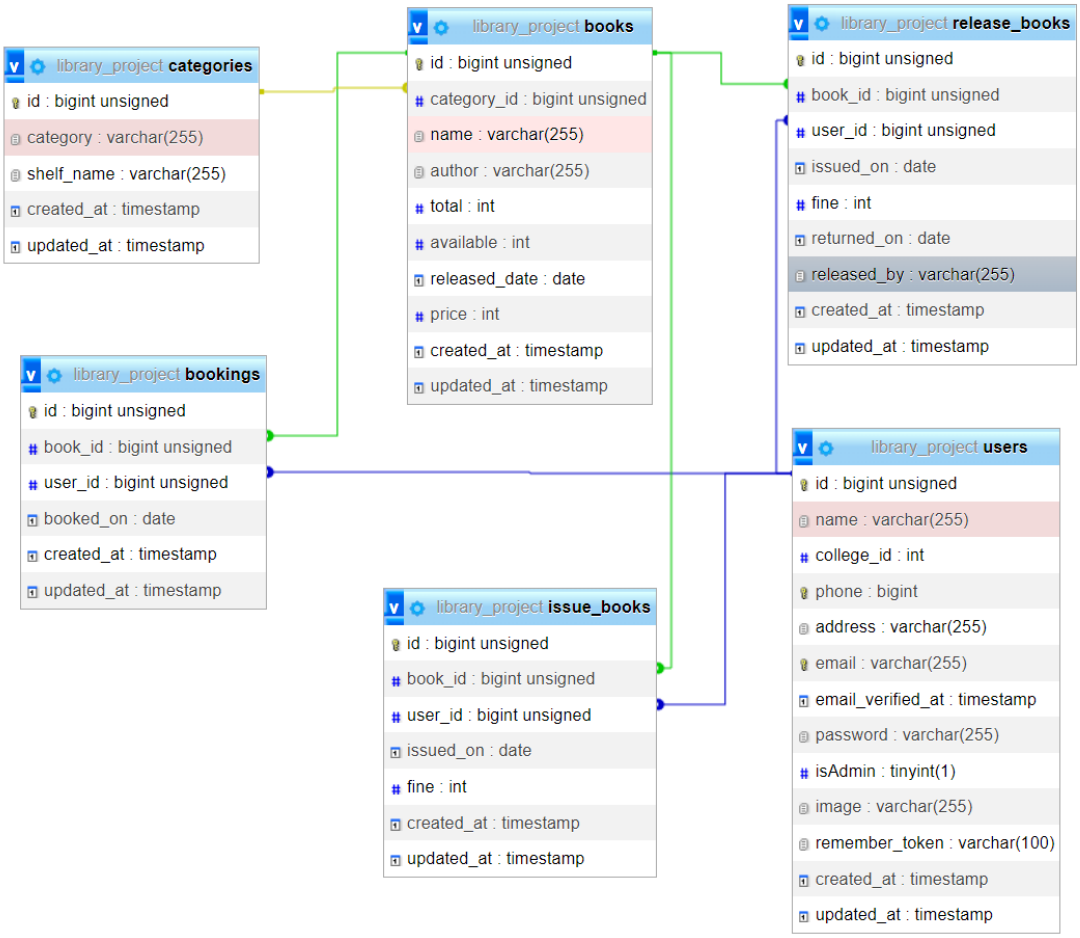


Fig 3.6.: Database Schema Design of LMS

3.2.3. Interface design

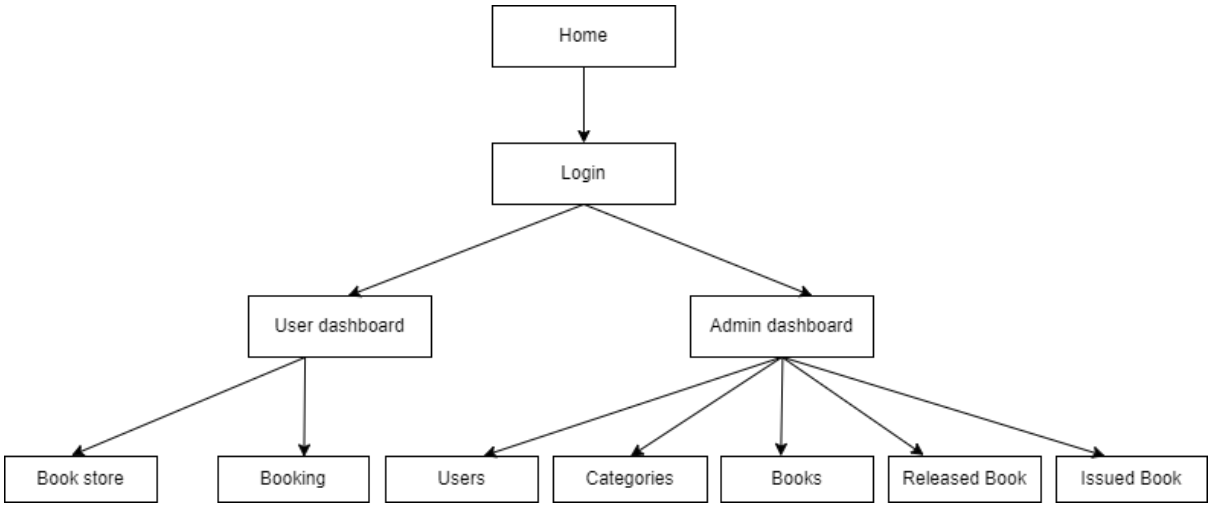


Fig 3.7.: Interface Design of LMS

3.2.4. Physical DFD

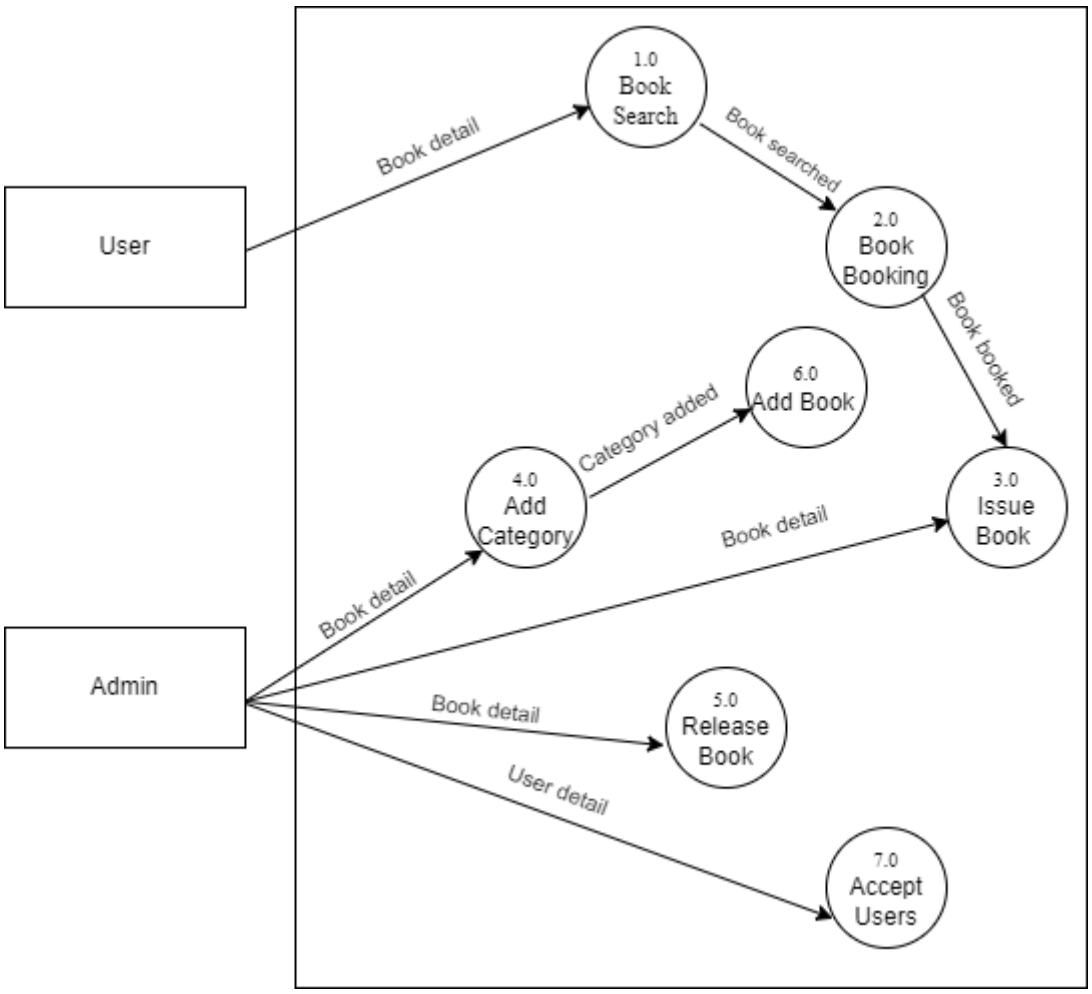


Fig 3.8.: Physical DFD of LMS

CHAPTER 4: IMPLEMENTATION AND TESTING

4.1 Implementation

4.1.1 Tools Used

I. draw.io: we use draw.io in order to create the context diagram, ER diagram, DFD, system design, etc.

II. VS code: We prefer using visual studio to code the program because of its flexibility and ease.

III. Programming language: we use php language to perform backend operation and Laravel as a framework.

IV. Database platform: MySQL is used as a database to store the information regarding books details, user details, etc.

4.1.2. Implementation details of module.

Login: This is the landing page of our system. In this module both Admin as well as user should first fill-up their details to get into the system.

Registration: The users should register first to get access into the system.

User dashboard: This is the user dashboard, when they login as a user they are directed into this module. In this module, user can search books, view books and book books.

Admin dashboard: This is the admin dashboard, when they login as an admin they are directed into this module. In this module, Admin can add users, add books according to category, view books, issue books to users, release the issued books, view user details and report on issued and returned books.

Book search: This module provides facility to both users and admin to search the available books in library.

Book booking: Users can interact with this module; they can book the available books in the library.

Category: Only admin can manipulate this module. Admin have authority to add the categories along with the shelf no. So, books can be managed in a systematic fashion.

Book: In this module admin have the authority to add books in library with respect to the category available.

Release book: This module is controlled by the admin. Admin can see and update the books that are released by the users.

Issue book: This module is only for the admin. Admin have the authority to issue book to the users with the help of identification details.

4.2. Testing

4.2.1. Test cases for unit testing:

Table 2: Unit Testing

Test case Id	Test cases	Input test data	Expected outcome	Result
TC_1	Test case for login with correct user name and wrong password	Username: admin@test.com Password:1234567	These credentials do not match the record	fail
TC_2	Test case for login with correct user name and correct password	Username: admin@test.com Password: password	Redirect to the admin dashboard	Pass
TC_3	Test case for login with correct user name and correct password	Username: user@test.com password: password	Redirect to the user dashboard	Pass
TC_4	Test cases for adding category by the admin	Category: Novel Shelf_name: N1	Category added successfully	Pass
TC_5	Test cases for adding book by the admin	Name: Harry potter Author: Jk Rowling Released_date: 2023-08-06 No of copies:3 Price:450	Book added successfully	Pass

4.2.2. Test cases for system testing

Table 3: Add book by the administrator.

Steps	Description
Step 1	Admin Login, Navigates to Admin dashboard
Step 2	Select the correct category. Category is selected
Step 3	Input the book title, author name, release date, availability, price.
Step 4	Click the submit button. Book is added and saved in database
Step 5	View the list of books clicking the book list button.

Table 4: Book booking by the user

Steps	Description
Step 1	User Login, navigates to user dashboard after the user is verified by admin.
Step 2	Select the correct category. Category is selected
Step 3	Input the book title, author name, release date, availability, price.
Step 4	Click the submit button. Book is added and saved in database
Step 5	View the list of books clicking the book list button.

CHAPTER 5: CONCLUSION AND FUTURE RECOMMENDATION

5.1. Lesson Learned/Outcome

Throughout this project journey, we've gained valuable insights. Our main goal of creating a user-friendly web-based system for managing books in library. Although the project is in its early stages, it has effectively met all our initial goals. From simplifying the registration process to seamlessly managing books, our system offers a straightforward and intuitive solution for finding books in library. Additionally, we learned an important lesson that even complex processes can be streamlined and made accessible through technology. As we move forward, we will continue to embrace technical solutions to enhance our operations and adapt to the ever-evolving workplace landscape.

5.2. Conclusion

In schools as well as in colleges, the Library Management System helps everyone find what they need easily. It's like a super organized tool that makes learning smoother for students and teachers. As things change, it changes too, keeping up with new ways of studying. Besides keeping things tidy, it encourages students to explore and learn more, making the learning experience even better. Basically, it's not just about managing information it's about sparking new ideas and making higher studies awesome.

5.3. Future Recommendations

Looking ahead, there are opportunities to make our Library Management System even better:

- **Automated Notifications:** We can set up the system to send automated messages for users regarding new books available in library.
- **Personalized User Experience:** Implement personalized dashboards, notifications, and reminders for users regarding due dates, upcoming events, or recommended reads.
- **Security Measures:** Strengthen security protocols to safeguard user data and prevent cyber threats, including measures like encryption, regular security audits, and user authentication

By considering these ideas, our Library Management System can continue to improve. It will not only handle user’s data effectively but also give us valuable insights and automate librarian tasks to run even smoother.

REFERENCES

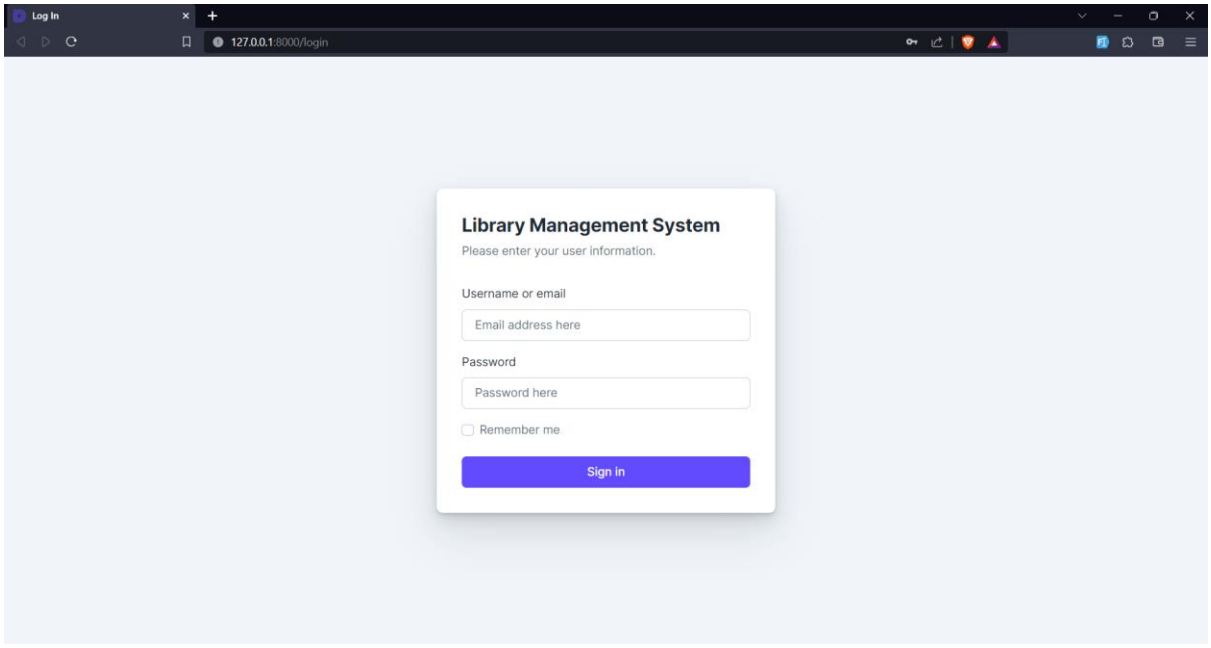
[1] RONG Mei, ZHANG Guang-Quan and LIU Yan, Design and Implement of Library System Based on Software Architecture and UML, 2018.

[2] QIAN Xiao-hua and Geng Cai-feng, The Building of Library Management System in B/S Structure Based on J2EE, 2019.

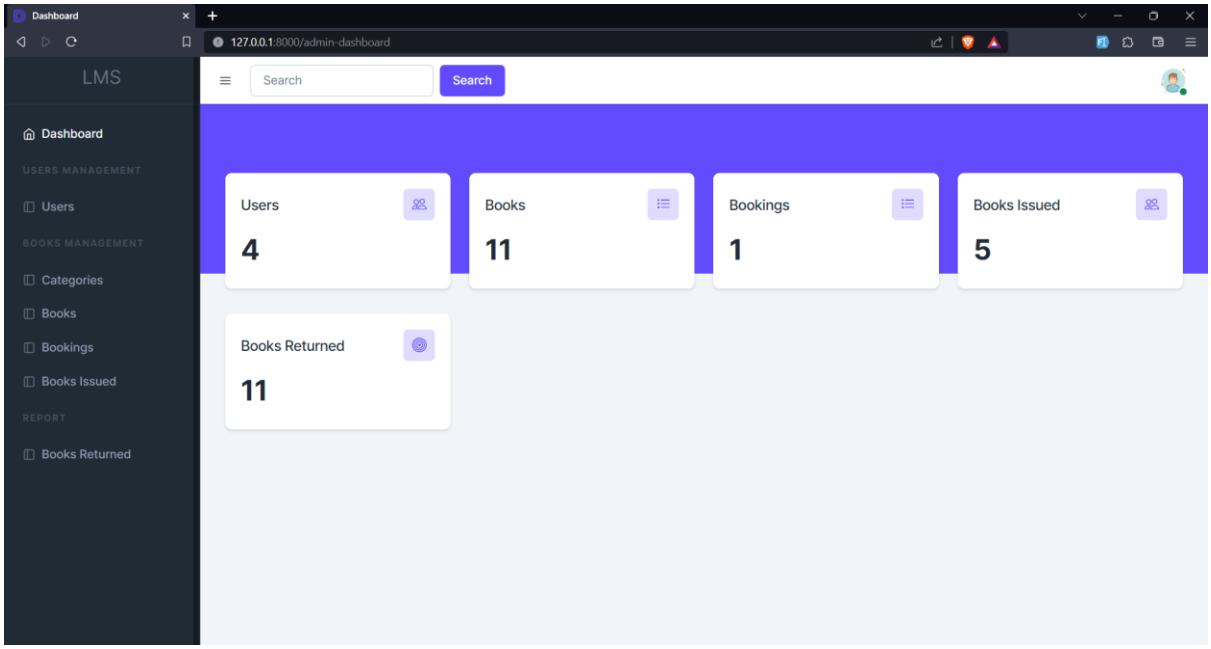
APPENDICES

Design Screenshots:

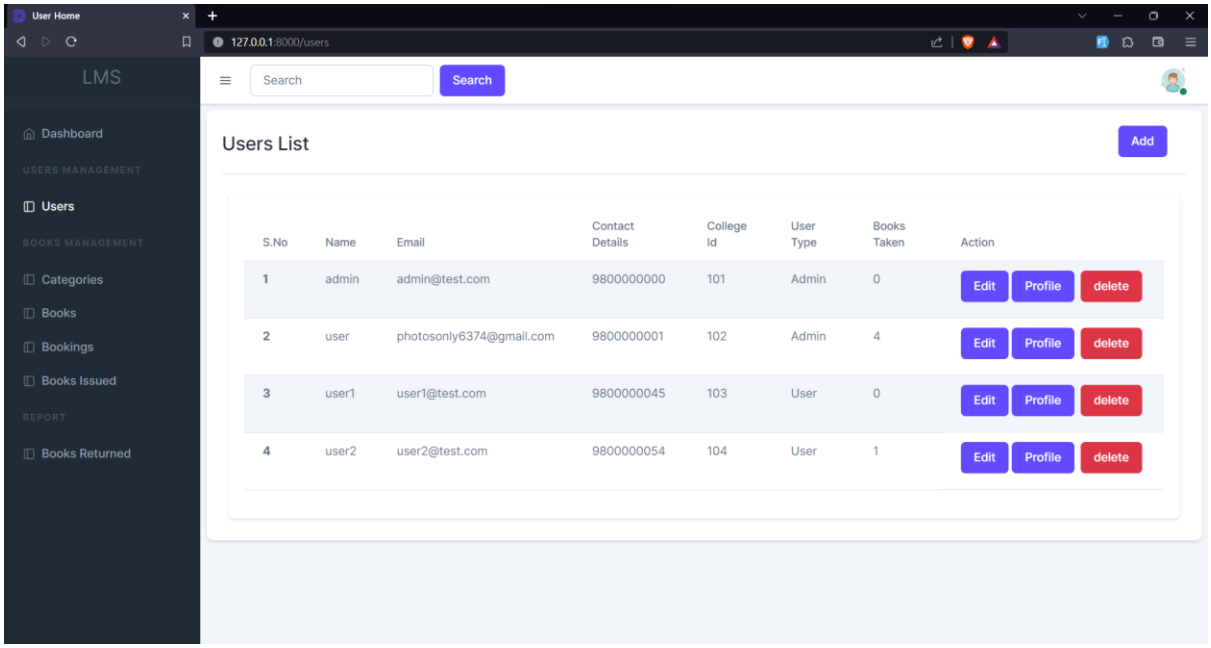
Login :



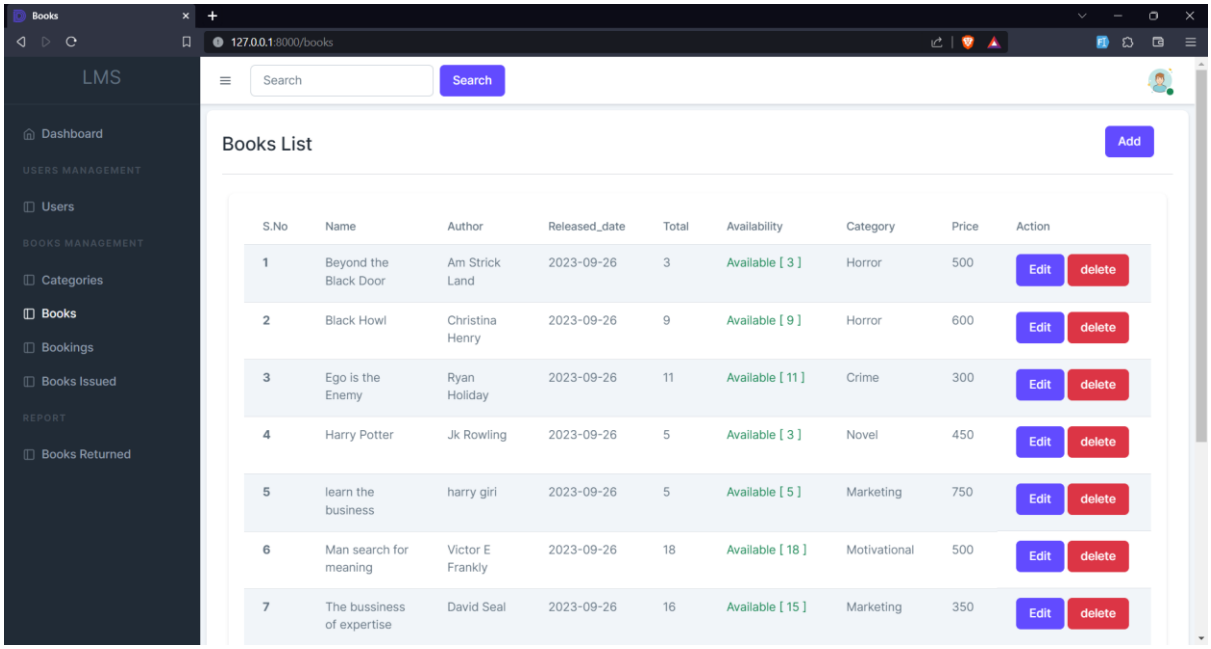
Admin Dashboard:



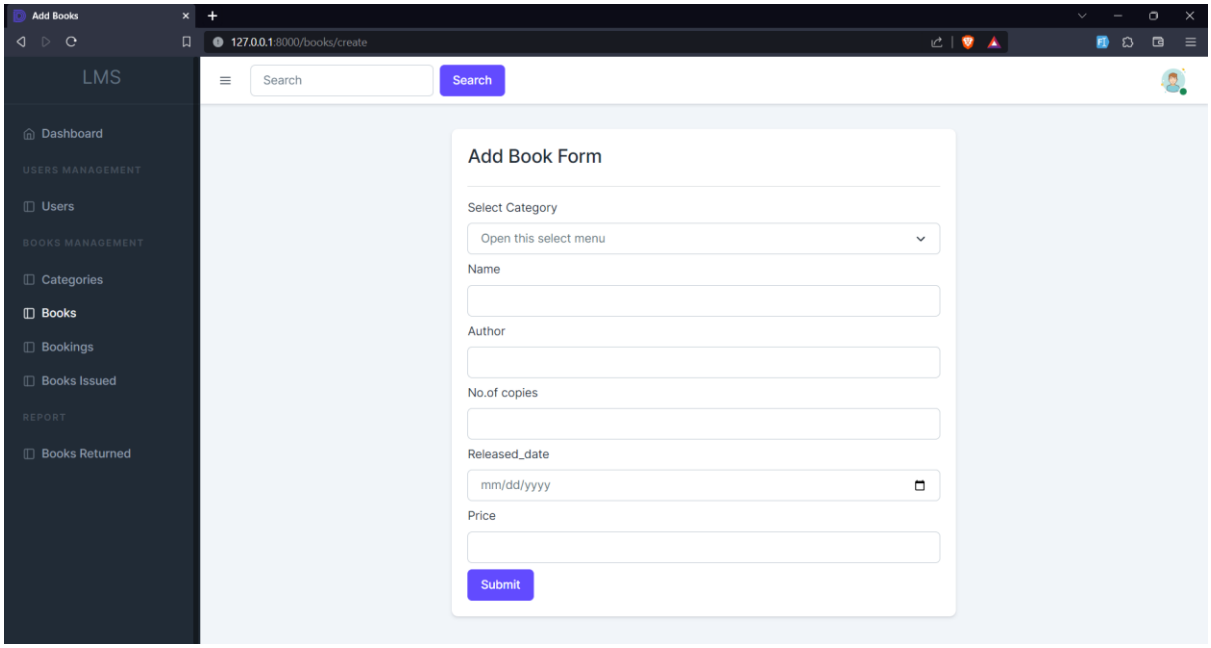
User List:



Book List:



Book Add Form



Source Code screenshots:

Routes

```php
Route::get('/',[DashboardController::class,'check']);

Route::group(['middleware' => ['auth']], function () {
    Route::get('/user-dashboard', [DashboardController::class, 'UserIndex'])->name('userDashboard');
    Route::get('/store', [HomeController::class, 'store'])->name('store');
    Route::get('/store/{id}', [HomeController::class, 'show'])->name('store.show');
    Route::get('logout', [LoginController::class, 'logout'])->name('logout');
    Route::post('/bookings', [BookingController::class, 'store'])->name('bookings.store');
});

Route::group(['middleware' => ['auth', 'isAdmin']], function () {
    Route::get('/admin-dashboard', [DashboardController::class, 'adminIndex'])->name('dashboard');

    Route::resource('categories', CategoryController::class);
    Route::resource('books', BookController::class);
    Route::resource('issue-books', IssueBookController::class);
    Route::resource('users', UserController::class);

    Route::get('/users/{id}/profile', [UserController::class, 'profile'])->name('users.profile');
    Route::get('/returned-books', [ReleaseBookController::class, 'index'])->name('returned-books.index');
    Route::get('/books-issue/release/{id}', [ReleaseBookController::class, 'release'])->name('issue-books.release');

    Route::get('/bookings', [BookingController::class, 'index'])->name('bookings.index');
    Route::get('/bookings/{id}/issue', [BookingController::class, 'issue'])->name('bookings.issue');
    Route::delete('/bookings/{id}', [BookingController::class, 'destroy'])->name('bookings.destroy');
});

Route::any('/{any}', [HomeController::class, 'error'])->where('any', '.*');
```

Dashboard:

```blade
@extends('layouts.app')
@section('title', 'Dashboard')
@section('content')
    <div class="bg-primary pt-10 pb-21"></div>

    <div class="container-fluid mt-n22 px-6">
        <div class="row">
            <div class="col-lg-12 col-md-12 col-12">
                <!-- Page header -->
                <div>
                    <div class="d-flex justify-content-between align-items-center">
                        <div class="mb-2 mb-lg-0">
                            {{-- <h3 class="mb-0  text-white">Projects</h3> --}}
                        </div>
                        <div>
                            {{-- <a href="#" class="btn btn-white">Create New Project</a> --}}
                        </div>
                    </div>
                </div>
            </div>
            <div class="col-xl-3 col-lg-6 col-md-12 col-12 mt-6">
                <!-- card -->
                <a href="{{ route('users.index') }}">
                    <div class="card ">
                        <!-- card body -->
                        <div class="card-body">
                            <!-- heading -->
                            <div class="d-flex justify-content-between align-items-center
                            mb-3">
```

```
                            <div>
                                <h4 class="mb-0">Users</h4>
                            </div>
                            <div class="icon-shape icon-md bg-light-primary text-primary
                            rounded-2">
                                <i class="bi bi-people fs-4"></i>
                            </div>
                        </div>
                        <!-- project number -->
                        <div>
                            <h1 class="fw-bold">{{ $users->count() }}</h1>
                            {{-- <p class="mb-0"><span class="text-dark
                            me-2">2<sspan>Completed</p> --}}
                        </div>
                    </div>
                </div>
            </a>
        </div>
        <div class="col-xl-3 col-lg-6 col-md-12 col-12 mt-6">
            <!-- card -->
            <a href="{{ route('books.index') }}">
                <div class="card ">
                    <!-- card body -->
                    <div class="card-body">
                        <!-- heading -->
                        <div class="d-flex justify-content-between align-items-center
                        mb-3">
                            <div>
```

```
                                <h4 class="mb-0">Books</h4>
                            </div>
                            <div class="icon-shape icon-md bg-light-primary text-primary
                            rounded-2">
                                <i class="bi bi-list-task fs-4"></i>
                            </div>
                        </div>
                        <!-- project number -->
                        <div>
                            <h1 class="fw-bold">{{ $books->count() }}</h1>
                            {{-- <p class="mb-0"><span class="text-dark me-2">28</
                            span>Completed</p> --}}
                        </div>
                    </div>
                </div>
            </a>
        </div>
        <div class="col-xl-3 col-lg-6 col-md-12 col-12 mt-6">
            <!-- card -->
            <a href="{{ route('bookings.index') }}">
                <div class="card ">
                    <!-- card body -->
                    <div class="card-body">
                        <!-- heading -->
                        <div class="d-flex justify-content-between align-items-center
                        mb-3">
                            <div>
                                <h4 class="mb-0">Bookings</h4>
                            </div>
```

```
                                <i class="bi bi-people fs-4"></i>
                            </div>
                        </div>
                        <!-- project number -->
                        <div>
                            <h1 class="fw-bold">{{ $booksIssued->count() }}</h1>
                            {{-- <p class="mb-0"><span class="text-dark me-2">1</
                            span>Completed</p> --}}
                        </div>
                    </div>
                </div>
            </a>
        </div>
        <div class="col-xl-3 col-lg-6 col-md-12 col-12 mt-6">
            <a href="{{ route('returned-books.index') }}">
                <div class="card ">
                    <div class="card-body">
                        <div class="d-flex justify-content-between align-items-center
                        mb-3">
                            <div>
                                <h4 class="mb-0">Books Returned</h4>
                            </div>
                            <div class="icon-shape icon-md bg-light-primary text-primary
                            rounded-2">
                                <i class="bi bi-bullseye fs-4"></i>
                            </div>
                        </div>
                        <div>
```

```
                            <div class="icon-shape icon-md bg-light-primary text-primary
                            rounded-2">
                                <i class="bi bi-list-task fs-4"></i>
                            </div>
                        </div>
                        <!-- project number -->
                        <div>
                            <h1 class="fw-bold">{{ $bookings->count() }}</h1>
                            {{-- <p class="mb-0"><span class="text-dark me-2">28</
                            span>Completed</p> --}}
                        </div>
                    </div>
                </div>
            </a>
        </div>
        <div class="col-xl-3 col-lg-6 col-md-12 col-12 mt-6">
            <!-- card -->
            <a href="{{ route('issue-books.index') }}">
                <div class="card ">
                    <!-- card body -->
                    <div class="card-body">
                        <!-- heading -->
                        <div class="d-flex justify-content-between align-items-center
                        mb-3">
                            <div>
                                <h4 class="mb-0">Books Issued</h4>
                            </div>
                            <div class="icon-shape icon-md bg-light-primary text-primary
                            rounded-2">
```

```
                        <div class="d-flex justify-content-between align-items-center
                        mb-3">
                            <div>
                                <h4 class="mb-0">Books Returned</h4>
                            </div>
                            <div class="icon-shape icon-md bg-light-primary text-primary
                            rounded-2">
                                <i class="bi bi-bullseye fs-4"></i>
                            </div>
                        </div>
                        <div>
                            <h1 class="fw-bold">{{ $booksReturned->count() }}</h1>
                            {{-- <p class="mb-0"><span class="text-success me-2">5%</
                            span>Completed</p> --}}
                        </div>
                    </div>
                </div>
            </a>
        </div>
    </div>
</div>
@endsection

```