LIBRARY MANAGEMENT ASSIGNMENT
(full stack)
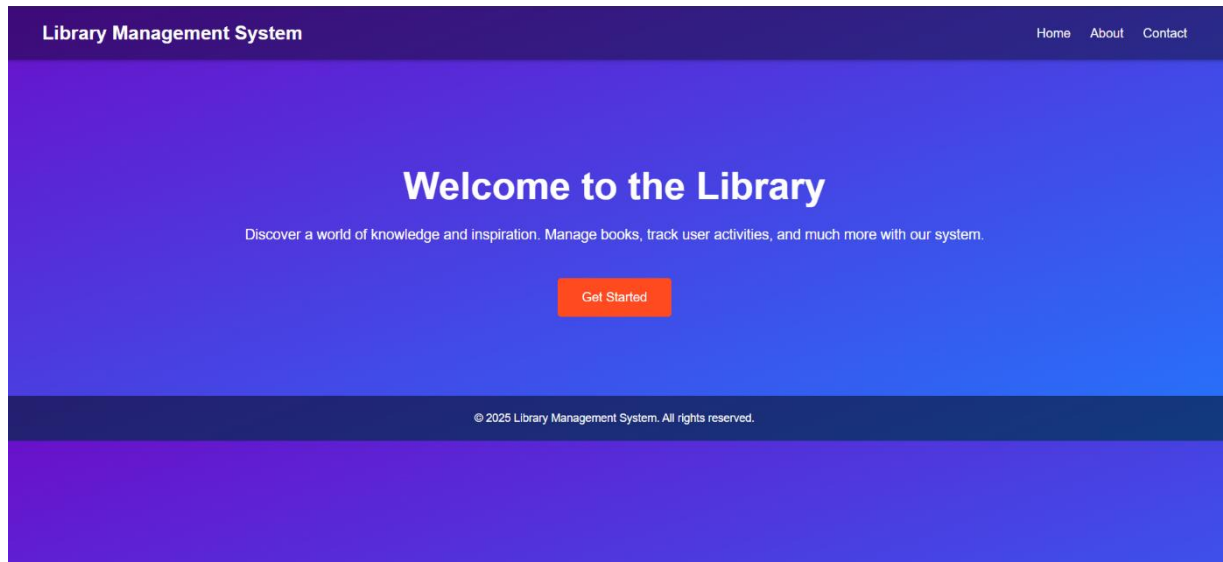

NAME: LAKSHYA GOTA
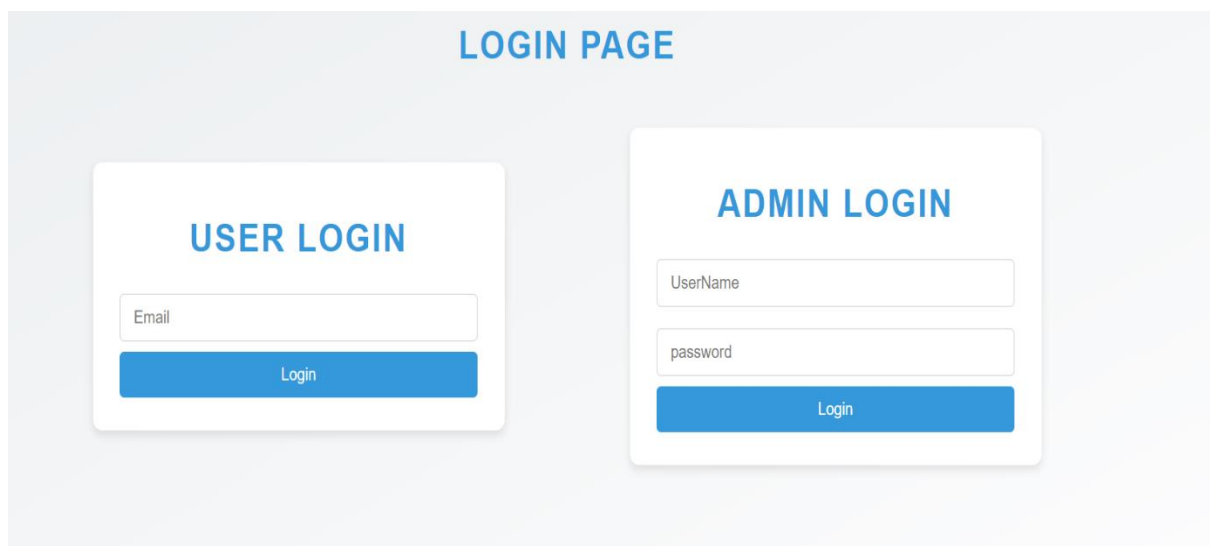
ROLL NO: 2K21/MC/95

COLLEGE: DELHI TECHNOLOGICAL UNIVERSITY

• HOME PAGE

Library Management System

Home   About   Contact

Welcome to the Library

Discover a world of knowledge and inspiration. Manage books, track user activities, and much more with our system.

Get Started

• LOGIN PAGE

LOGIN PAGE

USER LOGIN

Email

Login

ADMIN LOGIN

UserName

password

Login

• RETURN AND BORROW BOOKS PAGE

Welcome, Jane Smith!

Email: janesmith@example.com

Phone: 987-654-3210

Your Borrowed Books

Book Title	Author	Borrow Date	Return Date	Action
To Kill a Mockingbird	Harper Lee	Mon Jan 06 2025 00:00:00 GMT+0530 (India Standard Time)	Tue Jan 07 2025 00:00:00 GMT+0530 (India Standard Time)	
The Great Gatsby	F. Scott Fitzgerald	Mon Jan 06 2025 00:00:00 GMT+0530 (India Standard Time)	Tue Jan 07 2025 00:00:00 GMT+0530 (India Standard Time)	
The Catcher in the Rye	J.D. Salinger	Tue Jan 07 2025 00:00:00 GMT+0530 (India Standard Time)	Not Returned	Return
1984	George Orwell	Tue Jan 07 2025 00:00:00 GMT+0530 (India Standard Time)	Tue Jan 07 2025 00:00:00 GMT+0530 (India Standard Time)	
1984	George Orwell	Fri Jan 03 2025 00:00:00 GMT+0530 (India Standard Time)	Wed Jan 15 2025 00:00:00 GMT+0530 (India Standard Time)	
The Catcher in the Rye	J.D. Salinger	Tue Jan 07 2025 00:00:00 GMT+0530 (India Standard Time)	Tue Jan 07 2025 00:00:00 GMT+0530 (India Standard Time)	

Available Books

Book Title	Author	Availability	Action
1984	George Orwell	Available	Borrow
Rich Dad Poor Dad	Robert T. Kiyosaki	Available	Borrow
To Kill a Mockingbird	Harper Lee	Available	Borrow
The Great Gatsby	F. Scott Fitzgerald	Available	Borrow
1984	George Orwell	Available	Borrow
The Catcher in the Rye	J.D. Salinger	Available	Borrow

Logout

• ADD AND DELETE BOOKS

Books Management

All Books

ID	Title	Author	Publication Year	Availability	Action
2	1984	George Orwell	1949	Available	Delete
3	To Kill a Mockingbird	Harper Lee	1960	Available	Delete
4	The Catcher in the Rye	J.D. Salinger	1951	Not Available	Delete
5	The Great Gatsby	F. Scott Fitzgerald	1925	Available	Delete
6	1984	George Orwell	1949	Available	Delete
7	To Kill a Mockingbird	Harper Lee	1960	Not Available	Delete
8	The Catcher in the Rye	J.D. Salinger	1951	Available	Delete
10	Rich Dad Poor Dad	Robert T. Kiyosaki	1997	Available	Delete

Add New Book

Title

Author

Publication Year

Available

Add Book

DATABASE : PostgreSQL

- Tables (3)
  - books
  - transactions
  - users

- ## USERS TABLE

	id [PK] integer	name character varying (255)	email character varying (255)	phone character varying (15)
1	1	John Doe	johndoe@example.com	123-456-7890
2	2	Jane Smith	janesmith@example.com	987-654-3210
3	3	Sam Brown	sambrown@example.com	555-123-4567
4	4	Emily White	emilywhite@example.com	555-765-4321

- ## TRANSACTIONS TABLE

	id [PK] integer	book_id integer	user_id integer	borrow_date date	return_date date
1	2	2	2	2025-01-03	2025-01-15
2	3	3	3	2025-01-05	[null]
3	4	4	4	2025-01-07	[null]
4	6	3	2	2025-01-06	2025-01-07
5	7	5	2	2025-01-06	2025-01-07
6	8	2	2	2025-01-07	2025-01-07
7	9	4	2	2025-01-07	[null]
8	10	8	2	2025-01-07	2025-01-07

- ## BOOKS TABLE

	id [PK] integer	title character varying (255)	author character varying (255)	publication_year integer	availability boolean
1	2	1984	George Orwell	1949	true
2	3	To Kill a Mockingbird	Harper Lee	1960	true
3	4	The Catcher in the Rye	J.D. Salinger	1951	false
4	5	The Great Gatsby	F. Scott Fitzgerald	1925	true
5	6	1984	George Orwell	1949	true
6	7	To Kill a Mockingbird	Harper Lee	1960	false
7	8	The Catcher in the Rye	J.D. Salinger	1951	true
8	10	Rich Dad Poor Dad	Robert T. Kiyosaki	1997	true

Books Management System

Overview

The Books Management System is a web-based application that allows users to view, add, and delete books from a PostgreSQL database. It is built using Node.js, Express, and EJS for backend and frontend integration. The system provides a user-friendly interface for managing book details, making it suitable for small libraries or personal use.

Features

1. View All Books

- Displays all books stored in the database.
- Shows details such as:
  - Book ID
  - Title
  - Author
  - Publication Year
  - Availability

2. Add New Books

- Provides a form to input new book details:
  - Title
  - Author
  - Publication Year
  - Availability (Available/Not Available)
- Submits the data to the server to add the book to the database.

3. Delete Books

- Allows users to delete a book by its ID.
- Includes a confirmation prompt to prevent accidental deletions.

Technologies Used

Frontend

- **HTML**: Structure of the web pages.
- **CSS**: Styling for the application (inline and internal styles).

- **EJS**: Template engine for rendering dynamic content.

Backend

- **Node.js**: JavaScript runtime for server-side programming.

- **Express.js**: Web framework for handling routes and requests.

- **Body-Parser**: Middleware for parsing form data.

Database

- **PostgreSQL**: Relational database for storing book details.

System Requirements

- **Node.js** (v12 or higher)

- **PostgreSQL** (v12 or higher)

- Web browser

Database Schema

Table: books

Column	Type	Description
id	SERIAL	Primary Key
title	VARCHAR(255)	Book title
author	VARCHAR(255)	Author of the book
publication_year	INT	Year of publication
availability	BOOLEAN	Availability status

Sample SQL Query for Table Creation

```
CREATE TABLE books (
   id SERIAL PRIMARY KEY,
   title VARCHAR(255) NOT NULL,
   author VARCHAR(255) NOT NULL,
   publication_year INT NOT NULL,
   availability BOOLEAN NOT NULL
);
```

Installation and Setup

Step 1: Clone the Repository

git clone <repository_url>

cd project_folder

Step 2: Install Dependencies

npm install

Step 3: Configure Database

- Ensure PostgreSQL is running.

- Update the database connection details in the backend code:

- const db = new Pool({

- user: "your_username",

- host: "localhost",

- database: "your_database",

- password: "your_password",

- port: 5432,

- });

Step 4: Start the Server

node app.js

Step 5: Access the Application

Open your browser and navigate to:

http://localhost:3000/books

File Structure

project/

├── public/   # Static files (CSS, images, etc.)

├── views/   # EJS templates

| ├── books.ejs   # Main books page

├── app.js   # Backend server

├── package.json   # Project metadata and dependencies

Routes

1. /books (GET)

- Displays all books from the database.

2. /books/add (POST)

- Adds a new book to the database.

- Form data:

    o   Title

    o   Author

    o   Publication Year

    o   Availability

3. /books/delete/:id (POST)

- Deletes a book from the database by its ID.

Presentation Highlights

1. User Interface

- Clean and modern design with responsive layout.

- Intuitive forms for adding and deleting books.

2. Backend Integration

- Secure database interactions using parameterized queries.

- Real-time updates for managing books.

3. Database Management

- Robust schema for storing book details.

- Flexible operations to support adding and deleting records.

Future Enhancements

- Add authentication for admin access.

- Include pagination for large datasets.

- Add search functionality to filter books by title or author.

- Implement book editing functionality.

Conclusion

The Books Management System provides an efficient way to manage a library's inventory. Its simple yet powerful features make it ideal for small-scale library management or personal use. The project demonstrates effective use of modern web technologies and database management techniques.