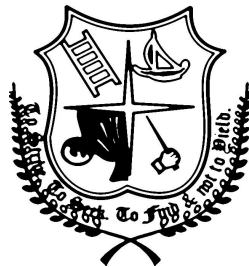


**ETHIRAJ COLLEGE FOR WOMEN  
(AUTONOMOUS)  
Chennai – 600 008**



**MASTER OF COMPUTER APPLICATIONS**

**December 2024**

**NAME : DHARSHINI U**  
**REG NO : 2413323037010**  
**SUB : COMPUTER LABORATORY-II:  
DATABASE MANAGEMENT  
SYSTEMS**  
**SUB CODE : MCA24/1P2/DMS**  
**SEM : I**

# PORTFOLIO MANAGEMENT SYSTEM

**ETHIRAJ COLLEGE FOR WOMEN  
(AUTONOMOUS)**

Chennai – 600 008.



**CERTIFICATE**

This is to certify that this is the bonafide record work carried out under my supervision in the Computer Laboratory Course “**COMP. LAB. II: DATABASE MANAGEMENT SYSTEMS**”, submitted to MCA Department, Ethiraj College for Women (Autonomous) by,

**DHARSHINI U**

**2413323037010**

as part of the course work leading to the award of the degree of

**MASTER OF COMPUTER APPLICATIONS**

**Faculty-In-Charge**

**Head of the Department**

Submitted For the Laboratory Examination at Ethiraj College For Women (Autonomous) on .....

**Examiner – 1**

**(Signature and Name of the Examiner)**

**Examiner – 2**

**(Signature and Name of the Examiner)**

## TABLE OF CONTENTS

<b>S.NO</b>	<b>CONTENTS</b>	<b>PAGE.NO</b>
<b>1.</b>	<b>INTRODUCTION</b>	<b>1</b>
	<b>1.1 SCOPE OF THE PROJECT</b>	<b>2</b>
<b>2.</b>	<b>HARDWARE/ SOFTWARE DESCRIPTION</b>	<b>4</b>
<b>3.</b>	<b>ABOUT THE SOFTWARE</b>	<b>6</b>
<b>4.</b>	<b>USE CASE DIAGRAM</b>	<b>8</b>
<b>5.</b>	<b>ER DIAGRAM</b>	<b>9</b>
<b>6.</b>	<b>MODULE DESCRIPTION</b>	<b>10</b>
<b>7.</b>	<b>SCREENSHOTS</b>	<b>13</b>
<b>8.</b>	<b>CODING</b>	<b>18</b>
<b>9.</b>	<b>FUTURE ENHANCEMENT</b>	<b>41</b>
<b>10.</b>	<b>REFERENCE</b>	<b>43</b>

# 1. INTRODUCTION

The **Portfolio Management System** is a comprehensive web application developed to simplify the process of organizing and presenting professional or creative work. Designed using a combination of HTML, CSS, PHP, and MySQL, the system enables users to securely manage their portfolios with an emphasis on usability, security, and aesthetic presentation. With the growing need for individuals and businesses to showcase their achievements and projects online, this system provides a centralized solution for creating and maintaining portfolios tailored to diverse needs.

The application revolves around four primary components: User authentication, portfolio items, portfolios, and categories, all interconnected through a relational database. The user authentication system ensures that only authorized individuals can access and modify portfolio data, providing a secure environment for sensitive information. The system allows users to create, edit, and organize portfolio items, each of which is enriched with detailed metadata, including titles, descriptions, URLs, and timestamps. Additionally, the ability to group portfolio items into broader portfolios offers a higher level of organization, ensuring clarity and coherence when presenting work to potential clients or collaborators. Categories further enhance this organization by allowing users to classify portfolio items for improved navigation.

The back-end of the system is powered by a robust MySQL database that efficiently manages relationships between users, portfolios, portfolio items, and categories. Each of these components is stored in structured tables, ensuring data integrity and easy retrieval. For instance, the user table stores login credentials, while the portfolio items table captures essential details about individual works, and the categories table facilitates classification. On the front-end, modern design principles using HTML and CSS ensure that the system is not only functional but also visually engaging and responsive across devices.

This project is particularly valuable for freelancers, professionals, and organizations that rely on showcasing their work online to attract clients or audiences. By providing a customizable and scalable platform, the Portfolio Management System caters to a broad range of users and industries. In an era where digital presence is crucial, this application empowers users to curate and display their work effectively, reinforcing their professional or creative identities.

This system is also designed with accessibility and cross-platform compatibility in mind. Its responsive design ensures optimal performance across various devices, including desktops, tablets, and smartphones. This adaptability makes it a versatile tool for professionals across industries, from designers and developers to writers and entrepreneurs, who require an impactful digital presence.

## 1.1 SCOPE OF THE PROJECT

The **Portfolio Management System** is a dynamic and user-centric application designed to facilitate the creation, management, and presentation of professional portfolios. The scope of the project includes the following key elements.

### Functional Scope:

#### i. User Authentication and Security:

- Secure login system to prevent unauthorized access.
- Storage of user credentials in the database with appropriate encryption mechanisms.

#### ii. Portfolio Management:

- Create, edit, and delete portfolios with attributes such as title, description, and creation date.
- Allow users to maintain multiple portfolios for different themes or purposes.

#### iii. Portfolio Items Management:

- Add, edit, delete, and view portfolio items.
- Attributes include item title, description, URL, and creation timestamp.
- Ensure all items are linked to their respective portfolios for better organization.

#### iv. Category Management:

- Create and manage categories to classify portfolio items.
- Allow assignment of items to specific categories for easier navigation and filtering.

### Technical Scope:

#### i. Frontend Development:

- Design and implement a responsive user interface using HTML and CSS.
- Ensure cross-platform compatibility across devices such as desktops, tablets, and mobile phones.

#### ii. Backend Development:

- Implement server-side logic using PHP for data processing and user interactions.
- Utilize MySQL as the database to store user information, portfolio data, and relationships between tables.

iii. **Database Structure:**

- Four main tables to manage data:

**Users:** Stores user credentials (username and password).

**Portfolios:** Contains portfolio titles, descriptions, and timestamps.

**Portfolio Items:** Captures details of individual items linked to portfolios.

**Categories:** Maintains a list of categories for item classification.

**Usability Scope:**

i. **Accessibility:**

- Design a system that is easy to use for individuals with varying levels of technical expertise.
- Ensure intuitive navigation and simplified processes for adding and managing portfolio content.

ii. **Customization:**

- Allow users to customize portfolio layouts and categorize items according to their preferences.

iii. **Future Scalability:**

- Provide a modular structure for adding new features, such as multimedia support, advanced analytics, or integration with third-party tools.

**Target Audience:**

- Professionals, freelancers, students, and businesses seeking a digital platform to showcase their work.
- Creative fields such as designers, developers, writers, photographers, and artists looking to maintain and present an organized portfolio.

This comprehensive scope ensures that the system caters to a broad range of user requirements while maintaining high standards of functionality, usability, and performance.

By addressing these key functionalities, the Portfolio Management System aims to deliver a robust, user-friendly platform that meets the needs of professionals looking to manage and showcase their work effectively.

## 2. HARDWARE/ SOFTWARE DESCRIPTION

The **Portfolio Management System** requires specific hardware and software components to ensure its proper development, deployment, and operation. Below is a detailed description of the required hardware and software.

### Hardware Requirements:

#### i. Development Environment

**Processor:** Intel Core i5.

**RAM:** Minimum 8 GB.

**Storage:** At least 256 GB SSD .

**Monitor:** Screen resolution of 1920x1080 or higher for better UI/UX design.

**Input Devices:** Standard keyboard and mouse.

**Network Connectivity:** High-speed internet connection (1 Gbps or higher).

#### ii. Client Environment

**Device:** Desktop, laptop, tablet, or smartphone.

**Browser Compatibility:** Supports modern web browsers such as Google Chrome, Mozilla Firefox, Safari, and Microsoft Edge.

#### iii. Optional Software

**Graphics Design Tools:** Adobe Photoshop, Figma, or Canva for UI design.

**Data Backup Tools:** Tools for scheduled backups, such as phpMyAdmin or custom scripts.

**Analytics Tools:** Google Analytics for tracking portfolio performance and audience engagement.



## Software Requirements

### i. Development Tools

**Operating System:** Windows 10/11, macOS, or Linux.

**Web Development Tools:**

- **Code Editor:** Visual Studio Code, Sublime Text
- **Version Control System:** Git (e.g., GitHub, GitLab, or Bitbucket).

### ii. Server-Side Software

**Web Server:** Apache or Nginx.

**Database Server:** MySQL 5.7 or higher.

**Server-Side Scripting:** PHP 7.4 or higher.

### iii. Frontend Development

**Languages:** HTML5, CSS3, JavaScript (optional for interactivity).

**Frameworks (optional):** Bootstrap for responsive design.

### iv. Backend Development

**Languages:** PHP for server-side logic and database interaction.

**Database Management:** MySQL for relational data storage.

### v. Testing Tools

**Browser Testing:** Google Chrome DevTools, Firefox Developer Edition.

**Debugging Tools:** Xdebug (PHP Debugger).

### 3. ABOUT THE SOFTWARE

The development of the **Portfolio Management System** involved the use of several key software tools and technologies to build a dynamic and interactive web application. Below is a detailed overview of the software used:

#### i. Visual Studio Code (VS Code)

**Purpose:** Code editor for writing, editing, and managing the project files.

**Key Features Utilized:**

- Syntax highlighting for JavaScript, PHP, HTML, and CSS.
- Extensions for PHP, MySQL, and JavaScript for enhanced functionality and debugging.
- Integrated terminal for running server commands and interacting with the database.
- Version control support for Git, aiding in version management and collaboration.

**Why Used:** Its lightweight nature, extensive extensions library, and powerful debugging tools make it a preferred choice for full-stack web development.

#### ii. XAMPP

**Purpose:** A local server environment for developing and testing the web application.

**Components Used:**

- **Apache:** Web server for hosting and serving PHP scripts and other resources.
- **MySQL:** Database server for storing and managing data related to users, portfolios, and categories.
- **phpMyAdmin:** Web interface for managing the MySQL database with SQL commands and GUI operations.

**Why Used:** XAMPP provides an all-in-one package, making it easy to set up a local development environment with essential tools like Apache, MySQL, and PHP.

#### iii. JavaScript

**Purpose:** Adding interactivity and client-side functionality to the web application.

**Usage in the Project:**

- Validating form inputs on the client side to ensure data integrity before submission.
- Enhancing user experience with dynamic elements such as dropdown menus and interactive buttons.
- Implementing asynchronous requests (AJAX) for smooth communication with the server without page reloads.

#### **iv. CSS (Cascading Style Sheets)**

**Purpose:** Styling the user interface to improve visual appeal and responsiveness.

**Usage in the Project:**

- Designing layouts and ensuring a consistent theme across all pages.
- Implementing responsive designs to ensure compatibility with various devices, including desktops, tablets, and smartphones.
- Adding hover effects, animations, and transitions to enhance user interactivity.

#### **v. PHP (Hypertext Preprocessor)**

**Purpose:** Server-side scripting for dynamic content and backend logic.

**Usage in the Project:**

- Handling form submissions and processing user inputs.
- Implementing user authentication (login and session management).
- Communicating with the MySQL database to perform CRUD (Create, Read, Update, Delete) operations.
- Generating dynamic web pages based on user actions and database queries.

#### **vi. MySQL**

**Purpose:** Database management system for storing, organizing, and retrieving data.

**Usage in the Project:**

- Storing data for user credentials, portfolios, portfolio items, and categories.
- Ensuring data relationships and integrity through table structures and foreign keys.
- Running queries to retrieve and manipulate data based on user interactions.

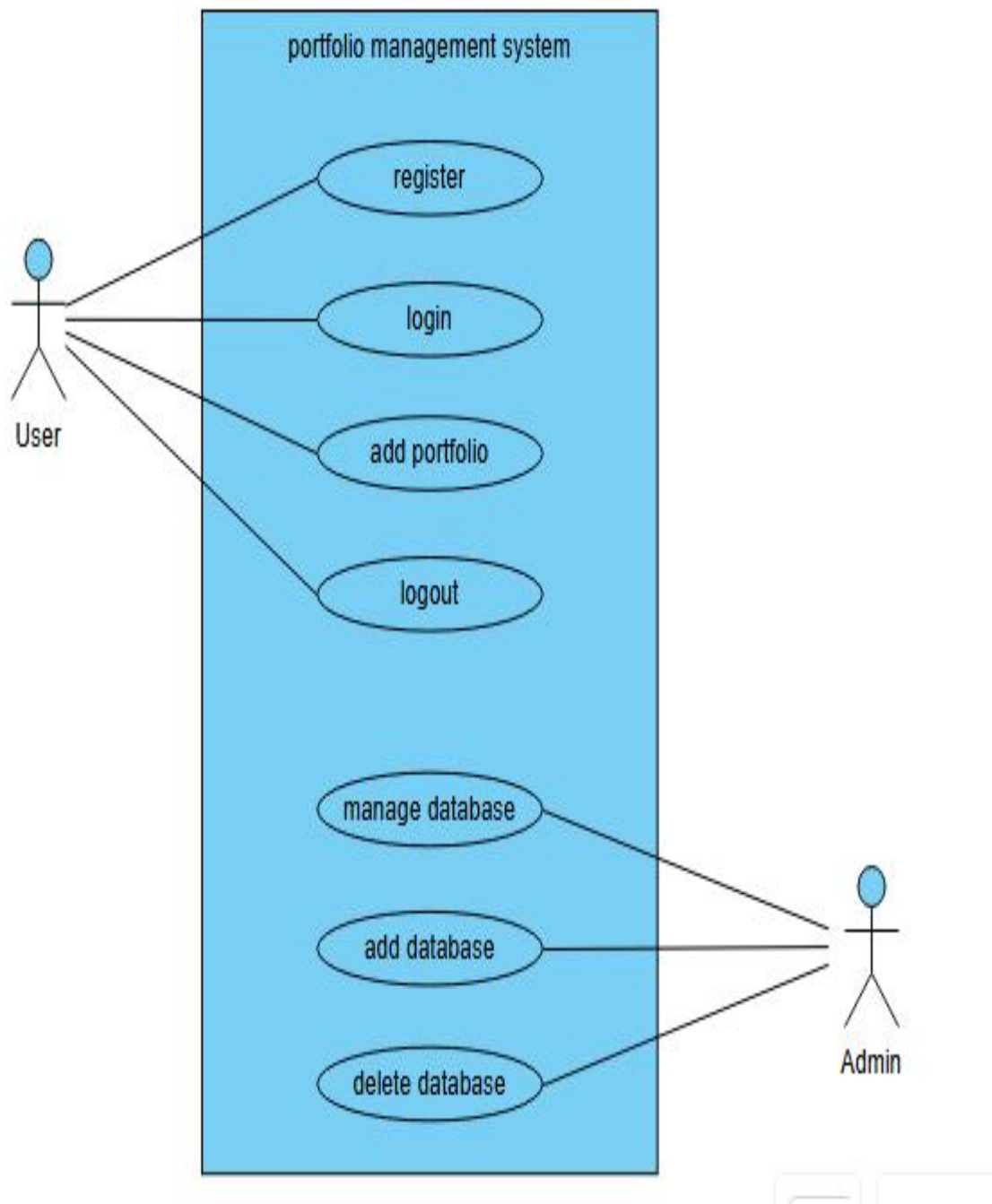
#### **vii. Apache**

**Purpose:** Web server for hosting and managing HTTP requests.

**Usage in the Project:**

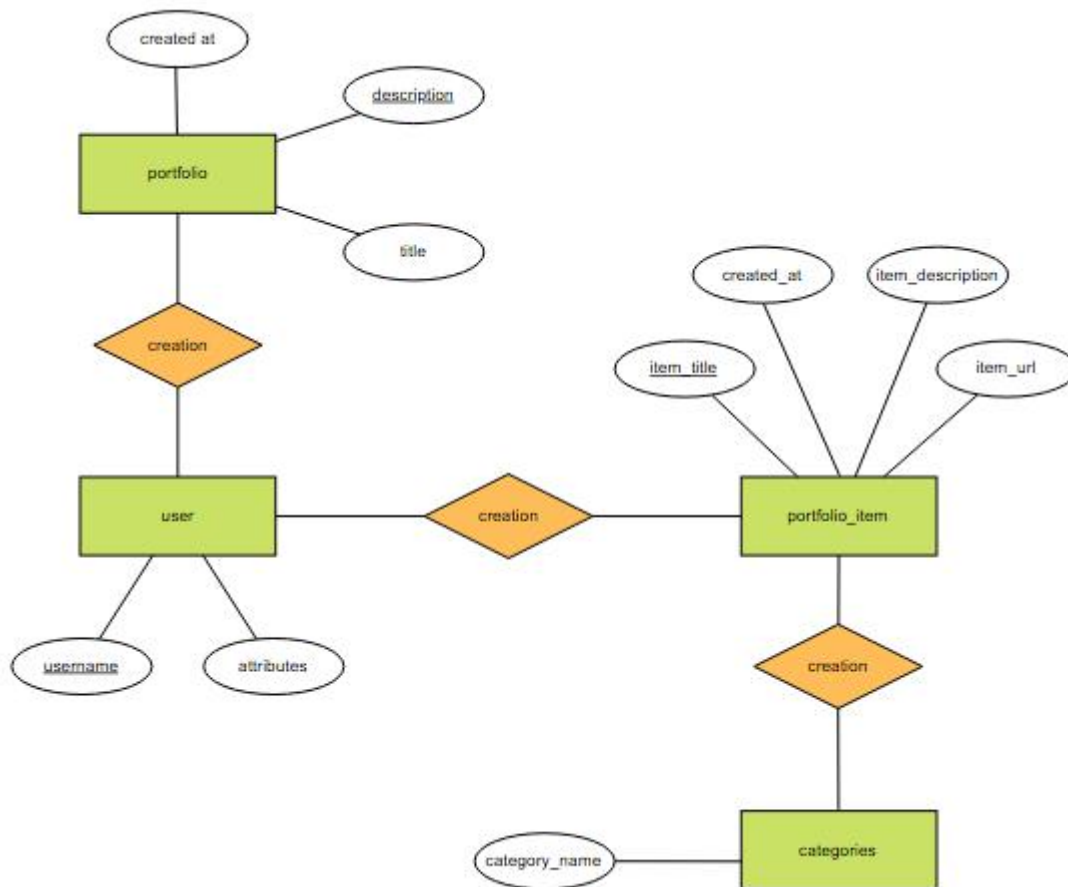
- Serving PHP scripts and web pages to the browser.
- Acting as the intermediary between the frontend (HTML, CSS, JavaScript) and the backend (PHP, MySQL).





## 4. USE CASE DIAGRAM



The use case diagram for your **Portfolio Management System**. It visually represents the interactions between the user and the system's functionalities.

## 5. ER DIAGRAM



Users		Portfolios		Categories		Portfolio_Items	
user_id 	INT	portfolio_id 	INT	category_id 	INT	item_id 	INT
username	VARCHAR	user_id	INT	category_name	VARCHAR	portfolio_id	INT
password	VARCHAR	title	VARCHAR			category_id	INT
		description	TEXT			item_title	VARCHAR
		created_at	TIMESTAMP			item_description	TEXT
						item_url	VARCHAR
						created_at	TIMESTAMP

The Entity-Relationship (ER) diagram for your Portfolio Management System. It represents the entities, attributes, and relationships in the system.

## 6. MODULE DESCRIPTION

The **Portfolio Management System** is divided into several key modules, each responsible for specific functionalities within the application. These modules work together to provide a seamless experience for managing and showcasing portfolio data. Below is a detailed description of each module:

### i. User Management Module

**Purpose:** Handles user authentication and access control.

**Features:**

- User registration with username and password.
- Secure login functionality using encrypted credentials.
- Session management to maintain user state during active sessions.
- Logout functionality to ensure security.

**Implementation:**

- **Frontend:** HTML forms styled with CSS.
- **Backend:** PHP scripts for validating credentials and creating sessions.
- **Database:** User details stored in the **Users** table with attributes:
  - username
  - password

### ii. Portfolio Management Module

**Purpose:** Enables users to create and organize portfolios.

**Features:**

- Add, view, update, and delete portfolios.
- Store portfolio attributes such as title, description, and creation date.

**Implementation:**

- **Frontend:** Form-based interface for creating and editing portfolios.
- **Backend:** PHP scripts for processing portfolio-related requests.
- **Database:** Portfolios stored in the **Portfolios** table with attributes:
  - title
  - description
  - created\_at

### iii. Portfolio Items Management Module

**Purpose:** Manages individual items within a portfolio.

**Features:**

- Add, view, edit, and delete portfolio items.
- Each item includes details such as title, description, URL, and creation timestamp.
- Link items to specific portfolios for better organization.

**Implementation:**

- **Frontend:** Interactive forms and item listing pages.
- **Backend:** PHP scripts for CRUD (Create, Read, Update, Delete) operations.
- **Database:** Portfolio items stored in the **Portfolio Items** table with attributes:
  - item\_title
  - item\_description
  - item\_url
  - created\_at

### iv. Category Management Module

**Purpose:** Organizes portfolio items into categories for better navigation and classification.

**Features:**

- Create, view, update, and delete categories.
- Assign portfolio items to specific categories.

**Implementation:**

- **Frontend:** Dropdown menus or selection fields for category management.
- **Backend:** PHP scripts for handling category assignments and modifications.
- **Database:** Categories stored in the **Categories** table with attributes:
  - category\_name

### v. Dashboard Module

**Purpose:** Provides an overview of the user's portfolio activities.

**Features:**

- Summary of portfolios, items, and categories.
- Navigation links to different modules.
- Real-time updates on recently added or modified content.

**Implementation:**

- **Frontend:** Dashboard layout styled with CSS for visual clarity.
- **Backend:** Dynamic data fetching using PHP and MySQL queries.

**vi. Search and Filter Module**

**Purpose:** Enhances user experience by allowing easy retrieval of portfolio items.

**Features:**

- Search for portfolio items by title or description.
- Filter items based on categories or creation date.

**Implementation:**

- **Frontend:** Search bars and dropdown filters.
- **Backend:** PHP scripts to handle search and filter queries.
- **Database:** Querying relevant tables for filtered results.

**vii. Responsive Design Module**

**Purpose:** Ensures the system is accessible across devices of varying screen sizes.

**Features:**

- Mobile-friendly layouts with flexible grids and responsive menus.
- Compatibility with desktops, tablets, and smartphones.

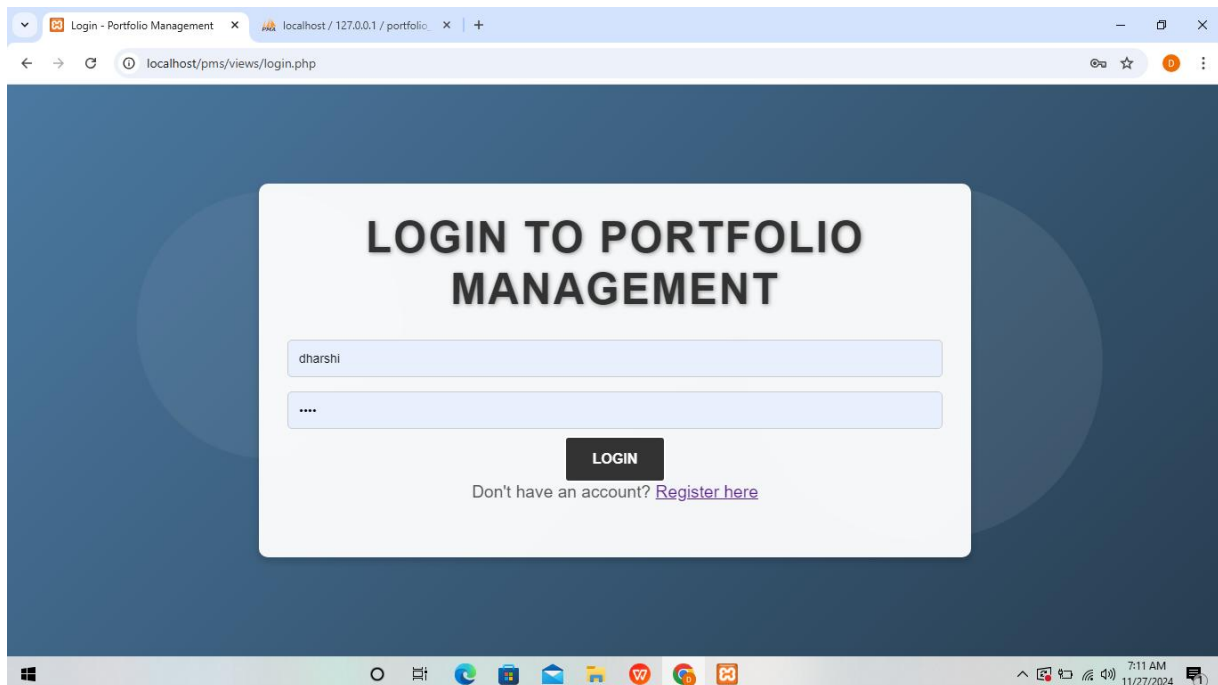
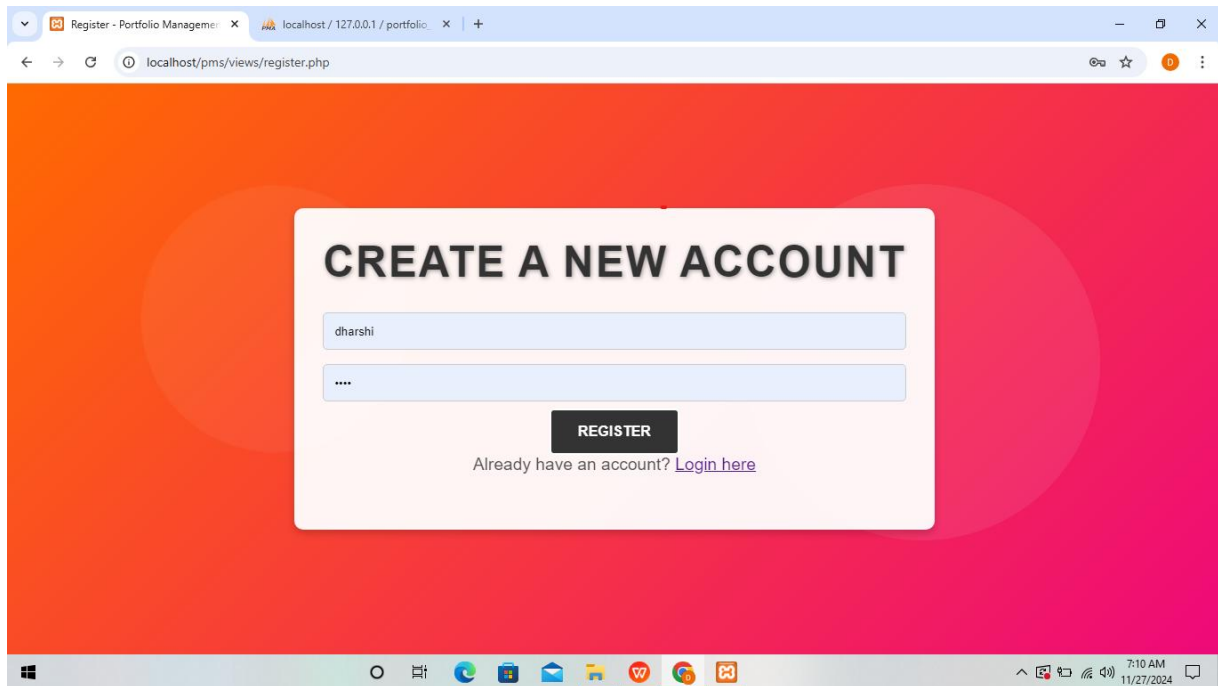
**Implementation:**

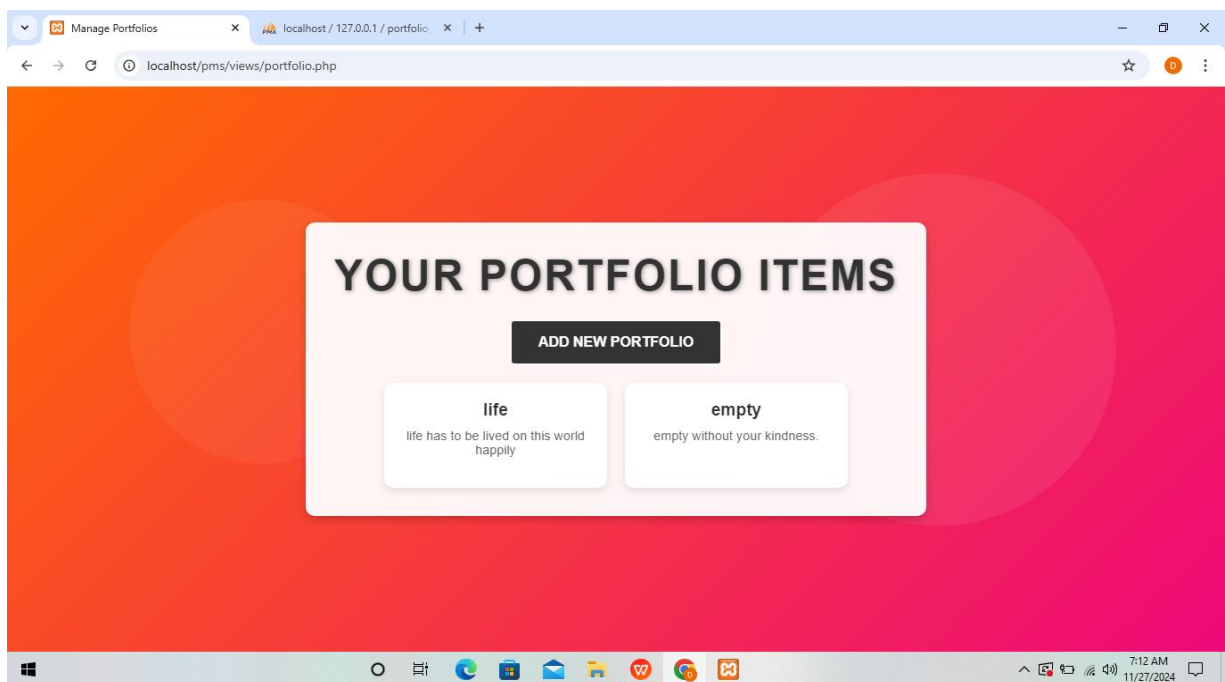
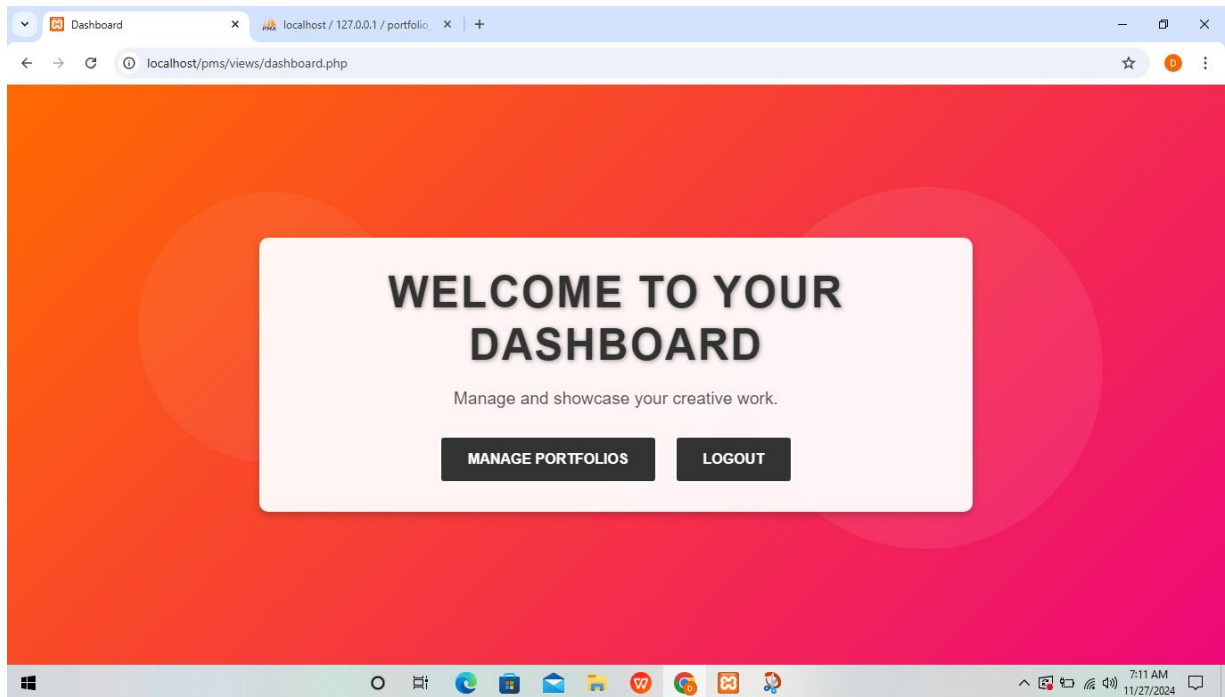
- CSS frameworks like Bootstrap for responsive design.
- Media queries for custom styling adjustments.

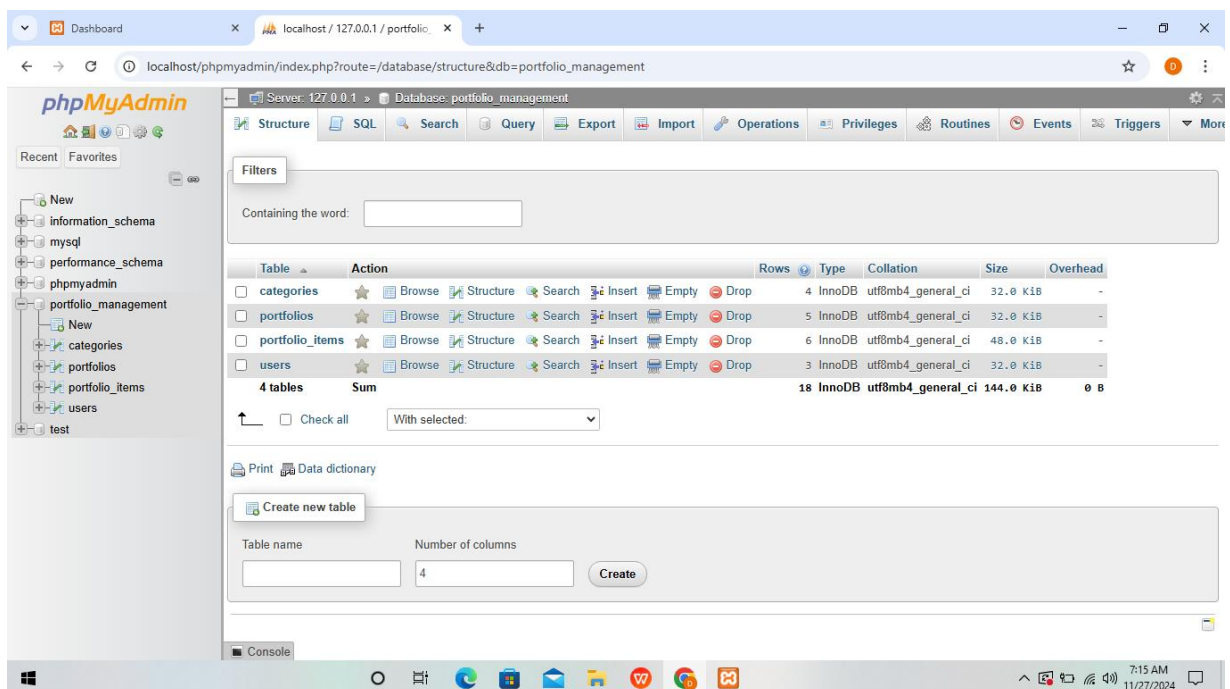
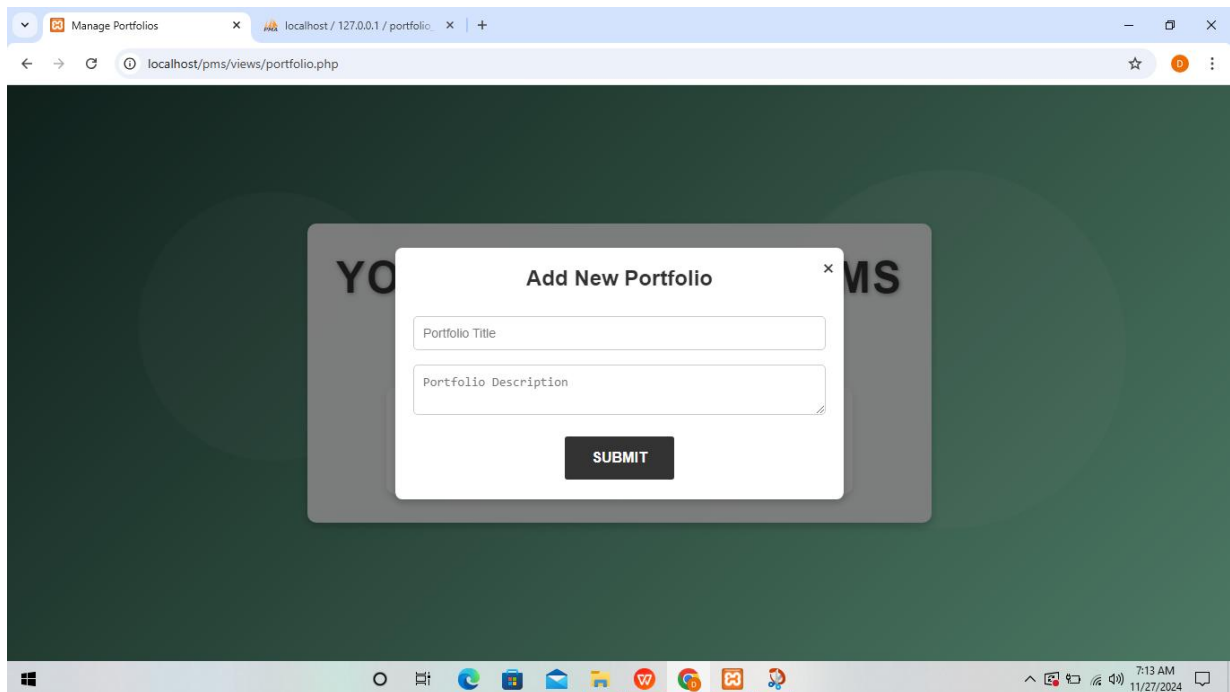
These modules collectively define the architecture of the **Portfolio Management System**, ensuring modularity, scalability, and ease of maintenance while delivering an efficient and user-friendly experience.



## 7. SCREENSHOTS







Dashboard | localhost / 127.0.0.1 / portfolio |

localhost/phpmyadmin/index.php?route=/sql&db=portfolio\_management&table=users&pos=0

Server: 127.0.0.1 | Database: portfolio\_management | Table: users

Showing rows 0 - 2 (3 total, Query took 0.0003 seconds.)

SELECT \* FROM `users`

Profiling [ Edit inline ] [ Edit ] [ Explain SQL ] [ Create PHP code ] [ Refresh ]

Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

Extra options

	user_id	username	password
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	1	john_doe	\$2y\$10\$Ej1YwG8XIQQONWICt/AXkeTzRcd7mFYh/FQW.e1u.BY...
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	2	jane_smith	\$2y\$10\$5y1pLe/MAj6szvQWukW1OeDoEtrR8E0crxSlPpGh5D...
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	3	dharshi	\$2y\$10\$X96hytm/6x5O/AyTh9ZRMorLAG9YCIpHfT7YtlUBVBU...

Check all | With selected: Edit Copy Delete Export

Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

Query results operations

Print Copy to clipboard Export Display chart Create view

Bookmark this SQL query

Console

Dashboard | localhost / 127.0.0.1 / portfolio |

localhost/phpmyadmin/index.php?route=/sql&pos=0&db=portfolio\_management&table=categories

Server: 127.0.0.1 | Database: portfolio\_management | Table: categories

Showing rows 0 - 3 (4 total, Query took 0.0003 seconds.)

SELECT \* FROM `categories`

Profiling [ Edit inline ] [ Edit ] [ Explain SQL ] [ Create PHP code ] [ Refresh ]

Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

Extra options

	category_id	category_name
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	4	E-commerce
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	2	Graphic Design
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	3	Photography
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	1	Web Development

Check all | With selected: Edit Copy Delete Export

Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

Query results operations

Print Copy to clipboard Export Display chart Create view

Console

Dashboard | localhost / 127.0.0.1 / portfolio

localhost/phpmyadmin/index.php?route=/sql&pos=0&db=portfolio\_management&table=portfolio\_items

Server: 127.0.0.1 | Database: portfolio\_management | Table: portfolio\_items

Showing rows 0 - 5 (6 total, Query took 0.0005 seconds.)

SELECT \* FROM `portfolio\_items`

Profiling [ Edit inline ] [ Edit ] [ Explain SQL ] [ Create PHP code ] [ Refresh ]

Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

Extra options

	item_id	portfolio_id	category_id	item_title	item_description	item_url	created_at
<input type="checkbox"/> Edit Copy Delete	1	1	1	Personal Website	A personal website built with HTML, CSS, and JavaS...	https://example.com/personal-website	2024-11-19 22:41:16
<input type="checkbox"/> Edit Copy Delete	2	1	1	E-commerce Store	An online store with a shopping cart system.	https://example.com/ecommerce-store	2024-11-19 22:41:16
<input type="checkbox"/> Edit Copy Delete	3	2	2	Logo Design	A minimalist logo designed for a startup.	https://example.com/logo-design	2024-11-19 22:41:16
<input type="checkbox"/> Edit Copy Delete	4	2	2	Poster Design	A poster designed for a local event.	https://example.com/poster-design	2024-11-19 22:41:16
<input type="checkbox"/> Edit Copy Delete	5	3	3	Nature Photography	Collection of nature-themed photographs.	https://example.com/nature-photos	2024-11-19 22:41:16
<input type="checkbox"/> Edit Copy Delete	6	3	3	Cityscapes	Urban landscapes and cityscapes.	https://example.com/cityscapes	2024-11-19 22:41:16

Check all | With selected: Edit Copy Delete Export

Console | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

Type here to search

Dashboard | localhost / 127.0.0.1 / portfolio

localhost/phpmyadmin/index.php?route=/sql&pos=0&db=portfolio\_management&table=portfolios

Server: 127.0.0.1 | Database: portfolio\_management | Table: portfolios

Showing rows 0 - 4 (5 total, Query took 0.0009 seconds.)

SELECT \* FROM `portfolios`

Profiling [ Edit inline ] [ Edit ] [ Explain SQL ] [ Create PHP code ] [ Refresh ]

Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

Extra options

	portfolio_id	user_id	title	description	created_at
<input type="checkbox"/> Edit Copy Delete	1	1	Web Development Projects	A collection of web development work.	2024-11-19 22:41:16
<input type="checkbox"/> Edit Copy Delete	2	1	Graphic Design Portfolio	Graphic design pieces.	2024-11-19 22:41:16
<input type="checkbox"/> Edit Copy Delete	3	2	Photography Portfolio	Photography highlights.	2024-11-19 22:41:16
<input type="checkbox"/> Edit Copy Delete	4	3	life	life has to be lived on this world happily	2024-11-20 07:28:26
<input type="checkbox"/> Edit Copy Delete	5	3	empty	empty without your kindness.	2024-11-27 06:54:44

Check all | With selected: Edit Copy Delete Export

Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

Query results operations

Print Copy to clipboard Export Display chart Create view

Console

Type here to search

## 8. CODING

**Source code:**

**Cascading style sheet:**

```
/* Full reset */

* {

    margin: 0;

    padding: 0;

    box-sizing: border-box;

}

body, html {

    height: 100vh;

    font-family: Arial, sans-serif;

    overflow: hidden;

}

.landing-container, .auth-container, .dashboard-container, .portfolio-container {

    position: relative;

    width: 100%;

    height: 100%;

    background: linear-gradient(135deg, #ff416c, #ff4b2b);

    animation: smoothGradient 20s linear infinite;

    display: flex;

    align-items: center;

    justify-content: center;

}
```

```

@keyframes smoothGradient {
  0% { background: linear-gradient(135deg, #ff416c, #ff4b2b); }
  25% { background: linear-gradient(135deg, #ff6a00, #ee0979); }
  50% { background: linear-gradient(135deg, #1f4037, #99f2c8); }
  75% { background: linear-gradient(135deg, #4b79a1, #283e51); }
  100% { background: linear-gradient(135deg, #ff416c, #ff4b2b); }
}

.overlay, .auth-overlay, .dashboard-overlay, .portfolio-overlay {
  color: #1a1a1a; /* Darker color for high contrast */
  text-align: center;
  position: relative;
  z-index: 2;
  max-width: 800px;
  padding: 2rem;
  background-color: rgba(255, 255, 255, 0.95);
  border-radius: 10px;
  box-shadow: 0 4px 8px rgba(0, 0, 0, 0.2);
  animation: fadeIn 1s ease;
}

.overlay h1, .auth-overlay h2, .dashboard-overlay h1, .portfolio-overlay h1 {
  font-size: 3em;
  font-weight: bold;
  text-transform: uppercase;
  letter-spacing: 2px;
  line-height: 1.2;
}

```

```

margin-bottom: 20px;

color: #333; /* High contrast */

text-shadow: 1px 1px 5px rgba(0, 0, 0, 0.4); /* Subtle shadow for readability */
}

.overlay p, .auth-overlay p, .dashboard-overlay p, .portfolio-overlay p {

font-size: 1.2em;

margin-bottom: 30px;

color: #555; /* Slightly darker for contrast */
}

.button-container, .auth-button-container {

display: flex;

gap: 20px;

justify-content: center;
}

.btn {

padding: 15px 30px;

font-size: 1em;

text-transform: uppercase;

font-weight: bold;

color: #ffffff;

background: #333; /* Dark background for high contrast */

border: 2px solid #ffffff;

border-radius: 5px;

text-decoration: none;

transition: all 0.3s ease;

```



```

}

.btn:hover {

    background: #ffffff;

    color: #333;

    transform: scale(1.1);

}

input[type="text"], input[type="password"] {

    width: 100%;

    padding: 12px;

    margin: 8px 0;

    border: 1px solid #ccc;

    border-radius: 5px;

    font-size: 14px;

}

form p {

    margin-top: 15px;

    font-size: 14px;

    color: #555;

}

/* Animated Elements (like orbs) */

.landing-container::before,

.landing-container::after,

.auth-container::before,

.auth-container::after,

.dashboard-container::before,

```

```

.dashboard-container::after,
.portfolio-container::before,
.portfolio-container::after {
    content: "";
    position: absolute;
    border-radius: 50%;
    opacity: 0.5;
    animation: float 15s ease-in-out infinite;
}

.landing-container::before,
.auth-container::before,
.dashboard-container::before,
.portfolio-container::before {
    width: 300px;
    height: 300px;
    background: rgba(255, 255, 255, 0.1);
    top: 20%;
    left: 10%;
    animation-delay: 0s;
}

.landing-container::after,
.auth-container::after,
.dashboard-container::after,
.portfolio-container::after {
    width: 400px;

```

```

height: 400px;

background: rgba(255, 255, 255, 0.15);

bottom: 20%;

right: 10%;

animation-delay: 5s;
}

@keyframes float {
    0% { transform: translateY(0) translateX(0); }
    50% { transform: translateY(-20px) translateX(20px); }
    100% { transform: translateY(0) translateX(0); }
}

@keyframes fadeIn {
    from { opacity: 0; transform: translateY(-10px); }
    to { opacity: 1; transform: translateY(0); }
}

/* Modal styling */

.modal {

    display: none; /* Hidden by default */

    position: fixed;

    top: 0;

    left: 0;

    width: 100%;

    height: 100%;

    background-color: rgba(0, 0, 0, 0.5); /* Dark semi-transparent background */

    align-items: center;

```

```
    justify-content: center;

    z-index: 1000;
}

.modal-content {

    background-color: #ffffff;

    padding: 20px;

    border-radius: 8px;

    max-width: 500px;

    width: 80%;

    box-shadow: 0 4px 8px rgba(0, 0, 0, 0.2);

    text-align: center;

    position: relative;
}

.close-btn {

    position: absolute;

    top: 10px;

    right: 10px;

    font-size: 24px;

    color: #333;

    cursor: pointer;
}

.modal h2 {

    margin-bottom: 20px;

    font-size: 1.5em;

    color: #333;
}
```

```
}

.modal input[type="text"],
.modal input[type="url"],
.modal textarea {
    width: 100%;
    padding: 10px;
    margin: 8px 0;
    border: 1px solid #ccc;
    border-radius: 5px;
    font-size: 14px;
}

.modal .btn {
    margin-top: 10px;
}

/* Portfolio Item Grid */

.portfolio-list {
    display: flex;
    flex-wrap: wrap;
    gap: 20px;
    margin-top: 20px;
    justify-content: center;
}

.portfolio-item {
    background-color: #ffffff;
    border-radius: 10px;
```

```
    box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1);

    padding: 20px;

    width: 250px;

    text-align: center;

    transition: transform 0.3s;
}

.portfolio-item:hover {

    transform: translateY(-5px);
}

.portfolio-item h3 {

    font-size: 1.2em;

    color: #333;

    margin-bottom: 10px;
}

.portfolio-item p {

    font-size: 0.9em;

    color: #666;

    margin-bottom: 15px;
}

.view-link {

    color: #ff416c;

    font-weight: bold;

    text-decoration: none;

    transition: color 0.3s;
}
```

```
.view-link:hover {  
    color: #ff4b2b;  
}
```

## **JAVASCRIPT:**

```
document.addEventListener('DOMContentLoaded', () => {  
  
    // Element selectors  
  
    const openModalBtn = document.getElementById('openModal'); // Button to open the  
    modal  
  
    const modal = document.getElementById('modal'); // Modal container  
  
    const closeModalBtn = document.getElementById('closeModal'); // Button to close the  
    modal  
  
    const addPortfolioForm = document.getElementById('addPortfolioForm'); // Add portfolio  
    form  
  
    const portfolioList = document.getElementById('portfolioList'); // Container for portfolios  
  
    /**  
    * Function to fetch and display all portfolios.  
    */  
  
    const fetchPortfolios = async () => {  
  
        try {  
  
            const response = await fetch('../fetch_portfolios.php'); // Fetch portfolios from the  
server  
  
            const portfolios = await response.json();  
  
            // Clear current list  
  
            portfolioList.innerHTML = "";  
  
            // Loop through portfolios and display them  
  
            portfolios.forEach((portfolio) => {  
  
                const portfolioDiv = document.createElement('div');
```

```

portfolioDiv.classList.add('portfolio-item');

portfolioDiv.innerHTML = `

  <h3>${portfolio.title}</h3>

  <p>${portfolio.description}</p>

  <ul>

    ${portfolio.items

      .map(

        (item) => `

          <li>

            <strong>${item.item_title}</strong> (${item.category_name ||
'Uncategorized'})

            <p>${item.item_description}</p>

            <a href="${item.item_url}" target="_blank">View</a>

          </li>

          ,

        )

      .join("")}

    </ul>

  `;

portfolioList.appendChild(portfolioDiv);

});

} catch (error) {

  console.error('Error fetching portfolios:', error);

}

};

```



```

/**
 * Function to open the modal.
 */

const openModal = () => {
    modal.style.display = 'flex';
};

/**
 * Function to close the modal.
 */

const closeModal = () => {
    modal.style.display = 'none';
};

/**
 * Function to handle the submission of the add portfolio form.
 */

const handleAddPortfolio = async (e) => {
    e.preventDefault(); // Prevent default form submission
    const formData = new FormData(addPortfolioForm);
    try {
        const response = await fetch('../add_portfolio.php', {
            method: 'POST',
            body: formData,
        });
        const result = await response.json();
    }

```

```

    if (result.status === 'success') {

        alert('Portfolio added successfully!');

        addPortfolioForm.reset(); // Reset the form

        closeModal(); // Close the modal

        fetchPortfolios(); // Refresh the portfolio list

    } else {

        alert(result.message);

    }

} catch (error) {

    console.error('Error adding portfolio:', error);

}

};

/**
 * Event Listeners
 */

openModalBtn.addEventListener('click', openModal); // Open modal on button click
closeModalBtn.addEventListener('click', closeModal); // Close modal on button click

// Close modal if the user clicks outside the modal content
window.addEventListener('click', (e) => {

    if (e.target === modal) {

        closeModal();

    }

});

```

```
addPortfolioForm.addEventListener('submit', handleAddPortfolio); // Handle form submission
```

```
/**  
 * Initial fetch of portfolios  
 */  
fetchPortfolios();  
});
```

## **DASHBOARD.php**

```
<!DOCTYPE html>  
  
<html lang="en">  
  
<head>  
  
  <meta charset="UTF-8">  
  
  <title>Dashboard</title>  
  
  <link rel="stylesheet" href="../css/styles.css">  
  
</head>  
  
<body>  
  
  <div class="dashboard-container">  
  
    <div class="dashboard-overlay">  
  
      <h1>Welcome to your Dashboard</h1>  
  
      <p>Manage and showcase your creative work.</p>  
  
      <div class="button-container">  
  
        <a href="portfolio.php" class="btn">Manage Portfolios</a>  
  
        <a href="../logout.php" class="btn">Logout</a>  
  
      </div>  
  
    </div>  
  
  </div>
```

```
</body>
```

```
</html>
```

## **LOGIN.php**

```
<!DOCTYPE html>
```

```
<html lang="en">
```

```
<head>
```

```
    <meta charset="UTF-8">
```

```
    <title>Login - Portfolio Management</title>
```

```
    <link rel="stylesheet" href="../css/styles.css">
```

```
</head>
```

```
<body>
```

```
    <div class="auth-container">
```

```
        <div class="auth-overlay">
```

```
            <h2>Login to Portfolio Management</h2>
```

```
            <form action="../login_user.php" method="POST">
```

```
                <input type="text" name="username" placeholder="Username" required>
```

```
                <input type="password" name="password" placeholder="Password" required>
```

```
                <button type="submit" class="btn">Login</button>
```

```
            </form>
```

```
            <p>Don't have an account? <a href="register.php">Register here</a></p>
```

```
        </div>
```

```
    </div>
```

```
</body>
```

```
</html>
```

## **INDEX.php**

```
<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <title>Welcome to Portfolio Management</title>

    <link rel="stylesheet" href="../css/styles.css">

</head>

<body>

    <div class="landing-container">

        <div class="overlay">

            <h1>Expand Your Creative Potential</h1>

            <p>Showcase and manage your work effortlessly.</p>

            <div class="button-container">

                <a href="login.php" class="btn">Login</a>

                <a href="register.php" class="btn">Register</a>

            </div>

        </div>

    </div>

</body>

</html>
```

## **REGISTER.php**

```
<html lang="en">

<head>

    <meta charset="UTF-8">
```

```

<title>Register - Portfolio Management</title>

<link rel="stylesheet" href="../css/styles.css">

</head>

<body>

  <div class="auth-container">

    <div class="auth-overlay">

      <h2>Create a New Account</h2>

      <form action="../register_user.php" method="POST">

        <input type="text" name="username" placeholder="Username" required>

        <input type="password" name="password" placeholder="Password" required>

        <button type="submit" class="btn">Register</button>

      </form>

      <p>Already have an account? <a href="login.php">Login here</a></p>

    </div>

  </div>

</body>

</html>

```

## **PORTFOLIO.php**

```

<!DOCTYPE html>

<?php

session_start();

// Redirect to login page if user is not logged in

if (!isset($_SESSION['user_id'])) {

    header("Location: login.php");

    exit;
}

```

```

}

// Include database connection

include '../db.php';

$user_id = $_SESSION['user_id'];

// Fetch all portfolios for the logged-in user

$result = $conn->query("SELECT * FROM Portfolios WHERE user_id = $user_id ORDER
BY created_at DESC");

?>

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Manage Portfolios</title>

    <link rel="stylesheet" href="../css/styles.css">

    <script src="../js/scripts.js" defer></script>

</head>

<body>

    <div class="portfolio-container">

        <div class="portfolio-overlay">

            <h1>Your Portfolio Items</h1>

            <!-- Button to open the modal -->

            <button id="openModal" class="btn">Add New Portfolio</button>

            <!-- Dynamic portfolio list -->

            <div id="portfolioList" class="portfolio-list">

                <?php while ($row = $result->fetch_assoc()) { ?>

```

```

<div class="portfolio-item">

    <h3><?php echo htmlspecialchars($row['title']); ?></h3>

    <p><?php echo htmlspecialchars($row['description']); ?></p>

    <p><small>Created on: <?php echo
htmlspecialchars($row['created_at']); ?></small></p>

    <!-- Optionally add a button or link for more actions -->

</div>

<?php } ?>

</div>

</div>

</div>

<!-- Modal for Adding New Portfolio -->

<div class="modal" id="modal">

    <div class="modal-content">

        <span class="close-btn" id="closeModal">&times;</span>

        <h2>Add New Portfolio</h2>

        <form id="addPortfolioForm">

            <input type="text" name="title" placeholder="Portfolio Title" required>

            <textarea name="description" placeholder="Portfolio Description"
required></textarea>

            <button type="submit" class="btn">Submit</button>

        </form>

    </div>

</div>

</body>

</html>

```



## **Dp.php**

```
<?php

// Check if a session is not already started

if (session_status() === PHP_SESSION_NONE) {

    session_start();

}

// Database connection

$conn = new mysqli('localhost', 'root', '', 'portfolio_management');

// Check connection

if ($conn->connect_error) {

    die("Connection failed: " . $conn->connect_error);

}

?>
```

## **Logout.php**

```
<?php

session_start();

session_unset();

session_destroy();

header("Location: views/index.php");

?>
```

## **Database.sql**

```
CREATE DATABASE portfolio_management;
```

```
USE portfolio_management;
```

```
-- Table: Users
```

```
CREATE TABLE Users (
```

```
    user_id INT AUTO_INCREMENT PRIMARY KEY,
```

```
    username VARCHAR(50) NOT NULL UNIQUE,
```

```
    password VARCHAR(255) NOT NULL
```

```
);
```

```
-- Table: Portfolios
```

```
CREATE TABLE Portfolios (
```

```
    portfolio_id INT AUTO_INCREMENT PRIMARY KEY,
```

```
    user_id INT,
```

```
    title VARCHAR(255) NOT NULL,
```

```
    description TEXT,
```

```
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
```

```
    FOREIGN KEY (user_id) REFERENCES Users(user_id) ON DELETE CASCADE
```

```
);
```

```
-- Table: Categories (for portfolio item categorization)
```

```
CREATE TABLE Categories (
```

```
    category_id INT AUTO_INCREMENT PRIMARY KEY,
```

```
    category_name VARCHAR(100) NOT NULL UNIQUE
```

```
);
```

-- Table: Portfolio\_Items (individual items in a portfolio)

```
CREATE TABLE Portfolio_Items (  
    item_id INT AUTO_INCREMENT PRIMARY KEY,  
    portfolio_id INT,  
    category_id INT,  
    item_title VARCHAR(255) NOT NULL,  
    item_description TEXT,  
    item_url VARCHAR(255),  
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,  
  
    FOREIGN KEY (portfolio_id) REFERENCES Portfolios(portfolio_id) ON DELETE  
    CASCADE,  
  
    FOREIGN KEY (category_id) REFERENCES Categories(category_id) ON DELETE SET  
    NULL  
);  
  
INSERT INTO Users (username, password)  
VALUES  
  
('john_doe', MD5('password123')),  
  
('jane_smith', MD5('securepass'));  
  
INSERT INTO Portfolios (user_id, title, description)  
VALUES  
  
(1, 'Web Development Projects', 'Showcase of my web development work.'),  
  
(1, 'Graphic Design Portfolio', 'A collection of graphic design pieces.'),  
  
(2, 'Photography Portfolio', 'Highlights of my photography journey.');  
  
INSERT INTO Categories (category_name)  
VALUES  
  
('Web Development'),
```

('Graphic Design'),

('Photography'),

('E-commerce');

INSERT INTO Portfolio\_Items (portfolio\_id, category\_id, item\_title, item\_description,  
item\_url)VALUES

(1, 1, 'Personal Website', 'A personal website built with HTML, CSS, and JavaScript.',  
'https://example.com/personal-website'),

(1, 1, 'E-commerce Store', 'An online store with a full shopping cart and checkout system.',  
'https://example.com/e-commerce-store'),

(2, 2, 'Logo Design', 'A minimalist logo designed for a startup.', 'https://example.com/logo-  
design'),

(2, 2, 'Poster Design', 'A poster designed for a local event.', 'https://example.com/poster-  
design'),

(3, 3, 'Nature Photography', 'A collection of nature-themed photographs.',  
'https://example.com/nature-photography'),

(3, 3, 'Cityscapes', 'Photographs of urban landscapes and cityscapes.',  
'https://example.com/cityscapes');

## 9. FUTURE ENHANCEMENT

The **Portfolio Management System** provides a robust foundation for managing and showcasing portfolios. However, there is significant potential for future enhancements to improve functionality, scalability, and user experience. Below are some suggestions for future upgrades:

### i. User Roles and Permissions

**Description:**

- Introduce different user roles, such as Admin, Editor, and Viewer, with varying levels of access and control.

**Benefits:**

- Improved security through restricted access to sensitive features.
- Supports collaboration by allowing multiple users to contribute to portfolios.

### ii. Drag-and-Drop Interface

**Description:**

- Implement a drag-and-drop feature for organizing portfolio items and categories.

**Benefits:**

- Enhances user experience with an intuitive interface.
- Speeds up the process of rearranging or categorizing items.

### iii. Advanced Search and Filtering

**Description:**

- Add advanced search options, such as filters for tags, keywords, and custom date ranges.

**Benefits:**

- Allows users to locate specific items or categories quickly.
- Improves system usability, especially for large portfolios.

### iv. Multimedia Support

**Description:**

- Enable users to upload images, videos, and documents for each portfolio item.

**Benefits:**

- Increases the richness of portfolio presentations.
- Expands the use cases for the system, accommodating photographers, videographers, and other multimedia professionals.

## **v. Analytics Dashboard**

### **Description:**

- Incorporate an analytics module to track portfolio performance, such as views, clicks, and engagement.

### **Benefits:**

- Provides valuable insights to users about their portfolio's reach and impact.
- Helps users identify popular items and optimize their content.

## **vi. Custom Themes and Templates**

### **Description:**

- Allow users to choose or create custom themes and layouts for their portfolios.

### **Benefits:**

- Adds personalization to portfolio designs.
- Attracts a broader audience by catering to diverse style preferences.

## **vii. Integration with Third-Party Tools**

### **Description:**

- Integrate with external platforms like social media (LinkedIn, Instagram), cloud storage (Google Drive, Dropbox), and content management systems.

### **Benefits:**

- Facilitates sharing of portfolio content across platforms.
- Makes it easier to import/export data from other tools.

## **viii. Email Notifications and Alerts**

### **Description:**

- Implement email notifications for key actions, such as successful login, new item additions, or password resets.

### **Benefits:**

- Keeps users informed about their account and activities.
- Enhances security through login and activity alerts.

By implementing these enhancements, the **Portfolio Management System** can evolve into a more versatile and user-friendly platform, meeting the growing demands of users and staying competitive in the digital portfolio management space.

## 10. REFERENCE

- 1) Meloni, J. C., & Kyrnin, J. (2017). *PHP, MySQL, & JavaScript All-in-One For Dummies*. Wiley.
- 2) Robbins, J. N. (2018). *Learning Web Design: A Beginner's Guide to HTML, CSS, JavaScript, and Web Graphics*. O'Reilly Media.
- 3) Freeman, E., & Robson, E. (2020). *Head First HTML and CSS: A Learner's Guide to Creating Standards-Based Web Pages*. O'Reilly Media.
- 4) Murach, J. (2017). *Murach's MySQL*. Mike Murach & Associates.
- 5) Oppel, A. (2009). *Databases Demystified: A Self-Teaching Guide*. McGraw-Hill.
- 6) Duckett, J. (2011). *HTML and CSS: Design and Build Websites*. John Wiley & Sons.
- 7) McFarland, D. S. (2015). *JavaScript and JQuery: Interactive Front-End Web Development*. Wiley.
- 8) Fowler, M. (2019). *Refactoring: Improving the Design of Existing Code*. Addison-Wesley.
- 9) Flanagan, D. (2020). *JavaScript: The Definitive Guide*. O'Reilly Media.
- 10) Ullman, L. (2019). *PHP and MySQL for Dynamic Web Sites: Visual QuickPro Guide*. Peachpit Press.
- 11) Nixon, R. (2021). *Learning PHP, MySQL & JavaScript: With jQuery, CSS & HTML5*. O'Reilly Media.
- 12) Official PHP Documentation: <https://www.php.net>
- 13) MySQL Documentation: <https://dev.mysql.com/doc/>
- 14) W3Schools: <https://www.w3schools.com/>
- 15) CSS Tricks: <https://css-tricks.com/>
- 16) Mozilla Developer Network (MDN): <https://developer.mozilla.org/>
- 17) Visual Studio Code: <https://code.visualstudio.com/>
- 18) XAMPP Documentation: <https://www.apachefriends.org/>
- 19) GitHub Documentation: <https://docs.github.com/>
- 20) SQL Tutorial: <https://sqlzoo.net/>
- 21) Cascading Style Sheets (CSS) Reference: <https://www.w3.org/Style/CSS/>
- 22) Xdebug for PHP Debugging: <https://xdebug.org/docs/>
- 23) Code Academy: Interactive tutorials: <https://www.codecademy.com/>

