

---

**Professor: Armando Ruggeri**  
**Student: Arya Khosravirad**  
**Matricola: 534 170**

---



Università  
degli Studi di  
Messina

Museum website

# CONTENTS

1. Introduction
2. Programming languages
3. Website
4. Codes
5. Database
6. Conclusion

# 1. Introduction:

---

The Museum Website project is designed to help manage museum operations. It enables visitors to book guided tours, register for events, and contact the museum . The website also provides tools for museum staff to manage schedules, visitor information, and administrative tasks.

## 2. Programming languages:

---

1. PHP: I used PHP to handle the backend of the museum website. It manages tasks such as connecting to the database and processing visitor input, like tour bookings or event registrations.

2. HTML: I used HTML to build the structure of the museum website. It organizes the content, including exhibition details, images, and visitor information, on each web page.

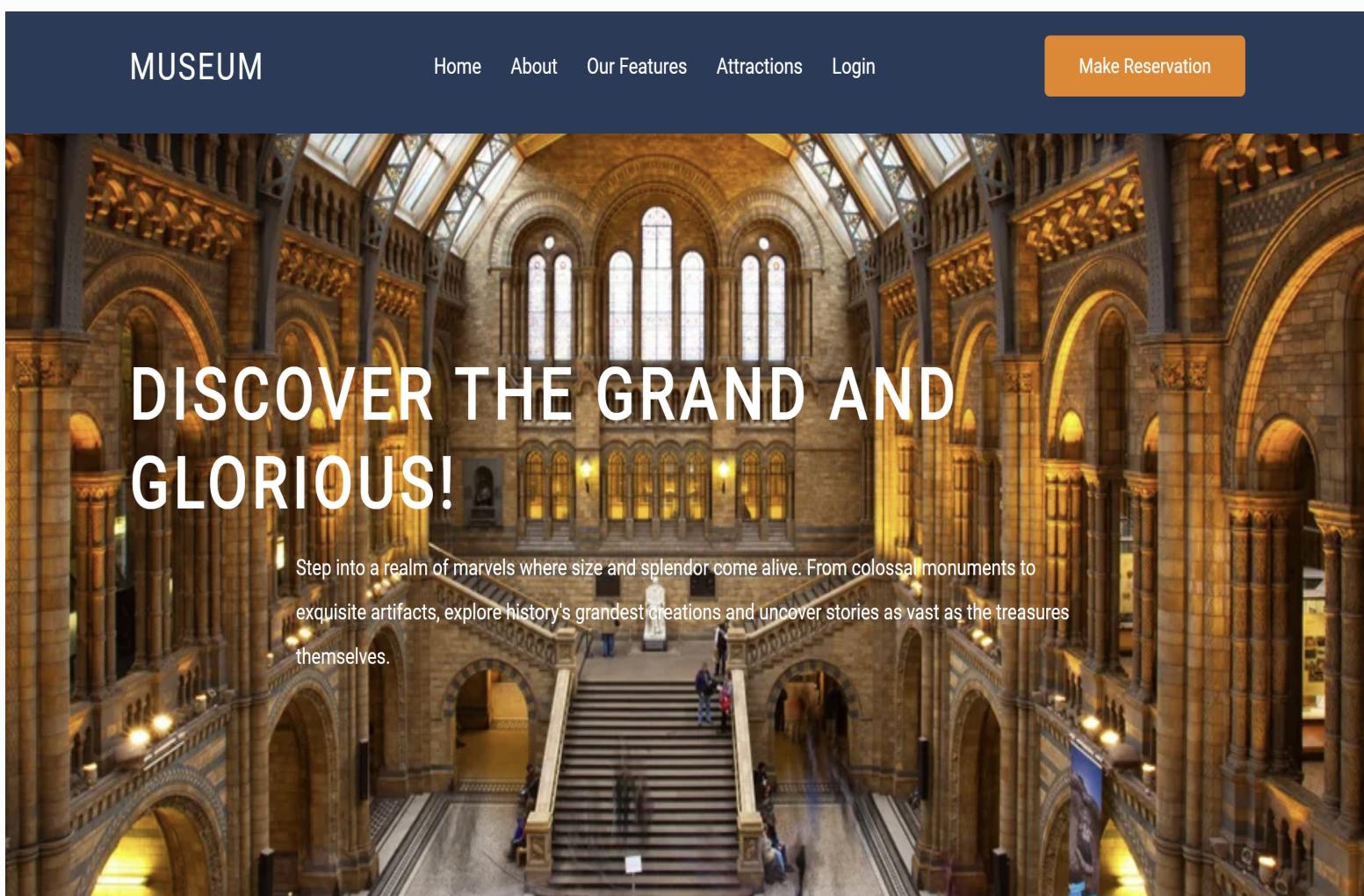
3. CSS: I used CSS to style the museum website. It controls the visual appearance, including colors, fonts, and layout, to create an appealing user experience.

4. JavaScript: I used JavaScript to add interactivity to the museum website.

# 3. Website

## 1. HOME PAGE:

1. The first part of the Reza Museum homepage has a simple and user-friendly design. It features a "Make Reservation" button, which makes it easy for users to book a date for their visit



## 2. Our features sections:

---

This section clearly showcases our features and highlights what you can do at our museum.

### Our Features



**Extensive Collections**  
Explore a wide range of artifacts, from ancient relics to modern masterpieces.



**Interactive Exhibits**  
Engage with history through touchscreens, virtual tours, and hands-on displays.



**Guided Tours**  
Learn fascinating stories behind the exhibits with our expert guides.



**Educational Workshops**  
Participate in classes and activities for all ages.



**Café & Gift Shop**  
Relax with a coffee or find unique souvenirs inspired by our collection.



**Event Spaces**  
Host memorable events in our stunning historical venue.

## 3. Attractions:

---

### Attractions



**Ancient Artifacts**  
Step into the world of history and explore artifacts from ancient civilizations.



**Masterpiece Exhibits**  
Admire the finest works of art, from classical paintings to modern masterpieces.



**Innovative Technology**  
Experience cutting edge technology with VR and interactive exhibits.

Here, I explain the attractions of our museum, which will appear when you hover your mouse over them.

## 4. About us:

---

Here you can  
read about  
Reza museum



About Us

### Museum Reza

We are dedicated to showcasing the world's most awe-inspiring wonders. From monumental achievements to exquisite artifacts, we celebrate human creativity and history. Our mission is to inspire, educate, and immerse visitors in the grandeur of past masterpieces.

Here is  
guidelines about  
how to book date

## 6. Make reservations:

---

### Make Reservation

Your Name

Your Phone Number

Your Email

mm / dd / yyyy



Message (Optional)

Reserve Now

Here allow user to  
fill out form which  
contains Name ,  
Email  
address,phone  
number , date and  
message (which is  
optional

## 7. Registration:

Before they can make reservations they must first register on website

The diagram illustrates the registration process in three main steps:

- Step 1: Personal Information**  
A screenshot of a registration form titled "Create Your Account". It includes fields for First Name and Last Name, both with placeholder text "Enter your first name" and "Enter your last name".
- Step 2: Contact Information**  
A screenshot of the same registration form showing the "Phone" and "Email Address" fields. The "Phone" field has the placeholder "Enter your phone number" and the "Email Address" field has the placeholder "Enter your email address".
- Step 3: Account Security**  
A screenshot of the registration form showing the "Address" field with placeholder "Enter your full address". Below it is the "Account Security" section with a "Password" field containing placeholder "Create a secure password" and a "Register Now" button.

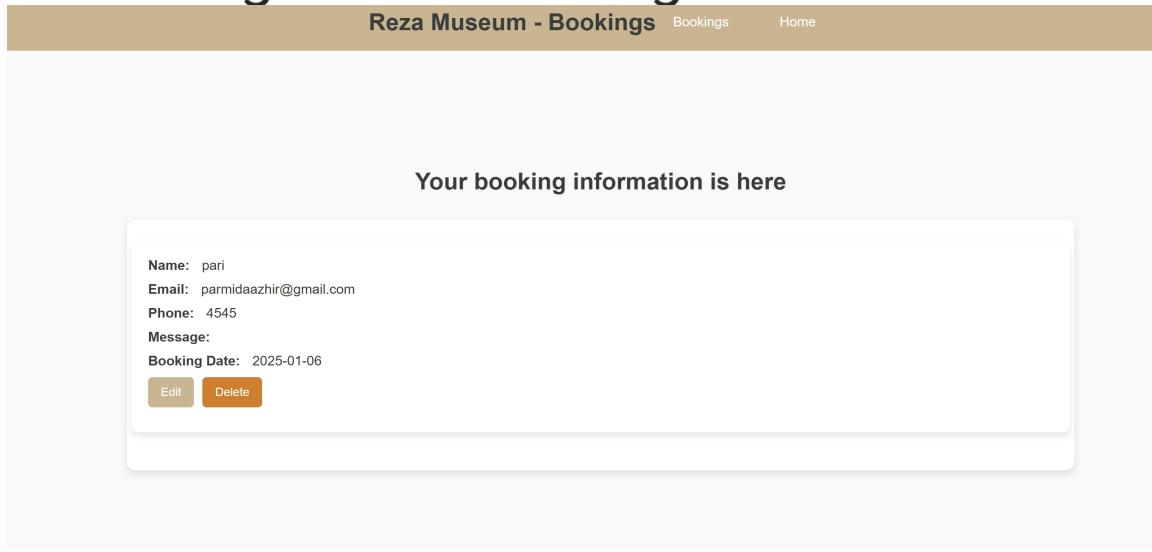
## 8. log in:

But if they already has an account they can easily log in **Reza Museum**

The login screen features a "Login" header and two input fields: "Email" and "Password", both with placeholder text. A large orange "Log In" button is centered below the fields. At the bottom, a link says "Don't have an account? Sign up here".

## 9. booking:

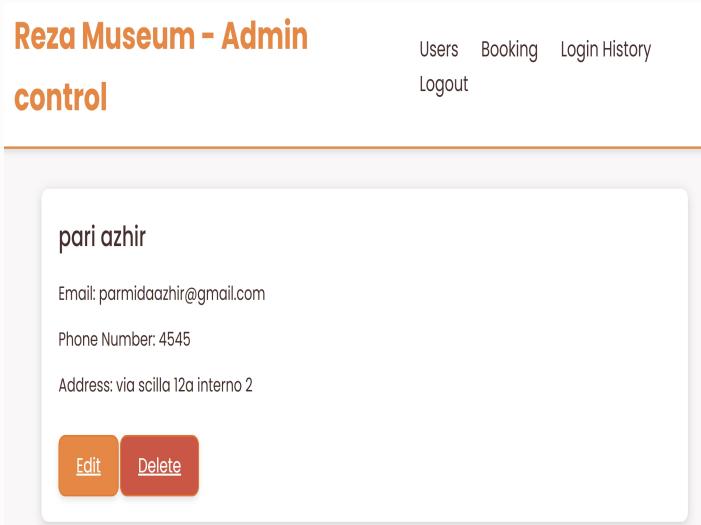
After user logged in and make reservation they can see and manage their booking here



The screenshot shows a booking information page with a header "Reza Museum - Bookings" and navigation links "Bookings" and "Home". The main content area displays booking details: Name: pari, Email: parmidaazhir@gmail.com, Phone: 4545, Message: (empty), Booking Date: 2025-01-06. Below the details are two buttons: "Edit" and "Delete".

## 10. admin page:

After i logged in  
as admin  
I can manage all  
the users  
booking here



The screenshot shows an admin control page for the Reza Museum. The top navigation bar includes "Reza Museum - Admin control", "Users", "Booking", "Login History", and "Logout". The main content area shows a user profile for "pari azhir" with details: Email: parmidaazhir@gmail.com, Phone Number: 4545, Address: via scilla 12a interno 2. Below the details are two buttons: "Edit" and "Delete".

## 11. Save on database :

After that every user make reservation or registration , their data will same on the museum database

Showing rows 0 - 0 (1 total, Query took 0.0009 seconds.)																															
SELECT * FROM `users`																															
<input type="checkbox"/> Profiling [ Edit inline ] [ Edit ] [ Explain SQL ] [ Create PHP code ] [ Refresh ]																															
<input type="checkbox"/> Show all		Number of rows: 25		Filter rows:		Search this table																									
Extra options																															
<table><thead><tr><th>← T →</th><th>id</th><th>firstname</th><th>lastname</th><th>phone</th><th>address</th><th>email</th><th>password</th></tr></thead><tbody><tr><td></td><td>2</td><td>pari</td><td>azhir</td><td>4545</td><td>12a interno</td><td>parmidaaazhir@gmail.com</td><td>\$2y\$12\$OmqSsLtik4HYCA90F0EHbeqyjhwHN8lJz39VHr71W...</td></tr><tr><td></td><td></td><td></td><td></td><td></td><td>via scilla</td><td></td><td></td></tr></tbody></table>								← T →	id	firstname	lastname	phone	address	email	password		2	pari	azhir	4545	12a interno	parmidaaazhir@gmail.com	\$2y\$12\$OmqSsLtik4HYCA90F0EHbeqyjhwHN8lJz39VHr71W...						via scilla		
← T →	id	firstname	lastname	phone	address	email	password																								
	2	pari	azhir	4545	12a interno	parmidaaazhir@gmail.com	\$2y\$12\$OmqSsLtik4HYCA90F0EHbeqyjhwHN8lJz39VHr71W...																								
					via scilla																										
<table><thead><tr><th><input type="checkbox"/></th><th><input type="checkbox"/> Edit</th><th><input type="checkbox"/> Copy</th><th><input type="checkbox"/> Delete</th><th>With selected:</th><th><input type="checkbox"/> Edit</th><th><input type="checkbox"/> Copy</th><th><input type="checkbox"/> Delete</th></tr></thead><tbody><tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr></tbody></table>								<input type="checkbox"/>	<input type="checkbox"/> Edit	<input type="checkbox"/> Copy	<input type="checkbox"/> Delete	With selected:	<input type="checkbox"/> Edit	<input type="checkbox"/> Copy	<input type="checkbox"/> Delete																
<input type="checkbox"/>	<input type="checkbox"/> Edit	<input type="checkbox"/> Copy	<input type="checkbox"/> Delete	With selected:	<input type="checkbox"/> Edit	<input type="checkbox"/> Copy	<input type="checkbox"/> Delete																								
<table><thead><tr><th><input type="checkbox"/></th><th><input type="checkbox"/> Check all</th><th>With selected:</th><th><input type="checkbox"/> Edit</th><th><input type="checkbox"/> Copy</th><th><input type="checkbox"/> Delete</th><th><input type="checkbox"/> Export</th></tr></thead><tbody><tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr></tbody></table>								<input type="checkbox"/>	<input type="checkbox"/> Check all	With selected:	<input type="checkbox"/> Edit	<input type="checkbox"/> Copy	<input type="checkbox"/> Delete	<input type="checkbox"/> Export																	
<input type="checkbox"/>	<input type="checkbox"/> Check all	With selected:	<input type="checkbox"/> Edit	<input type="checkbox"/> Copy	<input type="checkbox"/> Delete	<input type="checkbox"/> Export																									

## 4. Codes

```
<?php
require('DBconnection.php');

if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $name = mysqli_real_escape_string($conn, $_POST['name']);
    $phone = mysqli_real_escape_string($conn, $_POST['phone']);
    $email = mysqli_real_escape_string($conn, $_POST['email']);

    // Get the current date for the booking
    $booking_date = date('Y-m-d'); // Format: YYYY-MM-DD

    $message = !empty($_POST['message']) ? mysqli_real_escape_string($conn, $_POST['message']) : '';

    // Modify the SQL query to include the booking date
    $sql = "INSERT INTO book_reservation (name, email, phone, message, booking_date)
VALUES ('$name', '$email', '$phone', '$message', '$booking_date')";

    if (mysqli_query($conn, $sql)) {
        echo "<script>
            alert('Successfully booked in Reza Museum!');
            window.location.href='bookings.php';
        </script>";
    } else {
        echo "Error: " . $sql . "<br>" . mysqli_error($conn);
    }
}

mysqli_close($conn);
?>
```

## 1. Processes of booking :

It start on user dashboard where user fill out form, this form send data to book\_reservation , then book\_reservation will save details like name and ...

## 2. DB connection:

---

Connect website to  
MySQL database  
Using host ,  
username,databases  
name and password

```
src > DBconnection.php
1  <?php
2
3  $host = 'mysql';
4  $user = 'root';
5  $password = 'root';
6  $db = 'museum';
7  $conn = mysqli_connect($host, $user, $password, $db);
8
9  if (!$conn) {
10    echo "connection error --> " . mysqli_connect_error();
11 }
12
13 ?>
```

## 3 . Register processes:

---

Allow user to create an account  
After the user type they data , it will chek if  
the email exists or not and hashing the  
password for more security

```

<?php
session_start();

require('DBconnection.php');

function sanitize($data) {
    global $conn;
    return mysqli_real_escape_string($conn, htmlspecialchars($data));
}

function emailExists($email) {
    global $conn;
    $email = sanitize($email);
    $sql = "SELECT * FROM users WHERE email = '$email'";
    $result = $conn->query($sql);
    return $result->num_rows > 0;
}

function validatePassword($password) {
    return true;
}

if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $email = sanitize($_POST['email']);
    $password = sanitize($_POST['password']);
    $hashed_password = password_hash($password, PASSWORD_DEFAULT);

    if (emailExists($email)) {
        echo "Error: Email already exists. Please choose a different email.";
        exit();
    }

    if (!validatePassword($password)) {
        echo "Error: Password does not meet requirements.";
        exit();
    }

    $firstname = sanitize($_POST['firstname']);
}

```

1

```

$firstname = sanitize($_POST['firstname']);

$lastname = sanitize($_POST['lastname']);

$phone = sanitize($_POST['phone']);
$address = sanitize($_POST['address']);

$sql = "INSERT INTO users (firstname, lastname, phone, address, email, password)
        VALUES ('$firstname', '$lastname', '$phone', '$address', '$email', '$hashed_password')";

if ($conn->query($sql) === TRUE) {
    $_SESSION['email'] = $email;

    if (file_exists('login.php')) {
        header("Location: login.php");
    } else {
        echo "Error: Redirection file not found.";
    }
    exit();
} else {
    echo "Error: " . $sql . "<br>" . $conn->error;
}

$conn->close();
?>

```

2

## 4. Admin Dashboard:

After logging in , the admin can check the user details and edit them if needed

```

<?php
session_start();
require('DBconnection.php');

if (!isset($_SESSION['admin_id'])) {
    header('location: index.html');
    exit();
}

if (isset($_GET['delete_user_id'])) {
    $user_id = $_GET['delete_user_id'];
    $query = "DELETE FROM users WHERE id = $user_id";
    mysqli_query($conn, $query);

    $appointment_query = "DELETE FROM book_reservation WHERE email = (SELECT email FROM users WHERE id = $user_id)";
    mysqli_query($conn, $appointment_query);

    header('location: admin_dashboard.php#users');
    exit();
}

if (isset($_GET['delete_appointment_id'])) {
    $appointment_id = $_GET['delete_appointment_id'];
    $query = "DELETE FROM book_reservation WHERE id = $appointment_id";
    mysqli_query($conn, $query);

    header('location: admin_dashboard.php#book_reservation');
    exit();
}

if (isset($_GET['edit_user_id'])) {
    $user_id = $_GET['edit_user_id'];
    $query = "SELECT * FROM users WHERE id = $user_id";
    $result = mysqli_query($conn, $query);
    $user = mysqli_fetch_assoc($result);

    if ($_SERVER['REQUEST_METHOD'] == 'POST') {
        $firstname = $_POST['firstname'];
    }
}

```

1

```

$lastname = $_POST['lastname'];
$phone = $_POST['phone'];
$address = $_POST['address'];

$update_query = "UPDATE users SET firstname='$firstname', lastname='$lastname', phone='$phone', address='$address'";
if (mysqli_query($conn, $update_query)) {
    header('location: admin_dashboard.php#users');
    exit();
} else {
    echo "Error updating user: " . mysqli_error($conn);
}

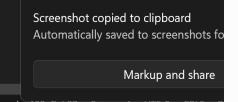
if (isset($_GET['edit_appointment_id'])) {
    $appointment_id = $_GET['edit_appointment_id'];
    $query = "SELECT * FROM book_reservation WHERE id = $appointment_id";
    $result = mysqli_query($conn, $query);
    $appointment = mysqli_fetch_assoc($result);

    if ($_SERVER['REQUEST_METHOD'] == 'POST') {
        $name = $_POST['name'];
        $phone = $_POST['phone'];

        $update_query = "UPDATE book_reservation SET name='$name', phone='$phone' WHERE id = $appointment_id";
        if (mysqli_query($conn, $update_query)) {
            header('location: admin_dashboard.php#book_reservation');
            exit();
        } else {
            echo "Error updating appointment: " . mysqli_error($conn);
        }
    }
}

```

3



2

```

// Query to fetch users
$user_query = "SELECT * FROM users";
$user_result = mysqli_query($conn, $user_query);

// Query to fetch book_reservation by joining on email
$appointment_query = "
    SELECT book_reservation.*, CONCAT(users.firstname, ' ', users.lastname) AS user_name
    FROM book_reservation
    LEFT JOIN users ON book_reservation.email = users.email";
$appointment_result = mysqli_query($conn, $appointment_query);

// Query to fetch login history
$login_query = "
    SELECT user_logins.*, CONCAT(users.firstname, ' ', users.lastname) AS user_name
    FROM user_logins
    LEFT JOIN users ON user_logins.user_id = users.id
    ORDER BY login_time DESC";
$login_result = mysqli_query($conn, $login_query);

```

## 5. Log in:

This code is for both admin and user ,  
After submitting data , it will check if data is set to  
be an admin or user , and if the data not in  
database, it will show message that user not found

```
rc > login.php
1  <?php
2  ob_start();
3  session_start();
4  require("DbConnection.php");
5
6 // Function to sanitize user input
7 function sanitize($data) {
8  global $conn;
9  return mysqli_real_escape_string($conn, htmlspecialchars($data));
10 }
11
12 if ($_SERVER["REQUEST_METHOD"] == "POST") {
13  $email = sanitize($_POST['email']);
14  $password = $_POST['password'];
15
16 // First, check if the credentials belong to an admin
17 $admin_query = "SELECT * FROM admin WHERE email = ?";
18 if ($admin_stmt = $conn->prepare($admin_query)) {
19  $admin_stmt->bind_param("s", $email);
20  $admin_stmt->execute();
21  $admin_result = $admin_stmt->get_result();
22
23  if ($admin_result->num_rows > 0) {
24   $admin_row = $admin_result->fetch_assoc();
25
26   if ($password === $admin_row['password']) { // Plain text password check for admin
27    $SESSION['admin_id'] = $admin_row['id'];
28    $SESSION['admin_email'] = $admin_row['email'];
29    header("location: admin_dashboard.php");
30    exit();
31   } else {
32    echo "<script>alert('Incorrect admin password.');" . window.location.href = 'login.php';</script>";
33   }
34  } else {
35   // If admin not found, proceed to check for user credentials
36   $user_query = "SELECT * FROM users WHERE email = ?";
37   if ($user_stmt = $conn->prepare($user_query)) {
38    $user_stmt->bind_param("s", $email);
39    $user_stmt->execute();
```

```
if ($user_result->num_rows > 0) {
  $user_row = $user_result->fetch_assoc();

  if (password_verify($password, $user_row['password'])) {
   // Record the login time in the user_logins table
   $login_time = date("Y-m-d H:i:s");
   $login_query = "INSERT INTO user_logins (user_id, login_time) VALUES (?, ?)";
   if ($login_stmt = $conn->prepare($login_query)) {
    $login_stmt->bind_param("is", $user_row['id'], $login_time);
    $login_stmt->execute();
    $login_stmt->close();
   }

   $_SESSION['user_id'] = $user_row['id'];
   $_SESSION['username'] = $user_row['firstname'] . ' ' . $user_row['lastname'];
   $_SESSION['email'] = $user_row['email'];
   header("location: user_dashboard.php");
   exit();
  } else {
   echo "<script>alert('Incorrect user password.');" . window.location.href = 'login.php';</script>";
  }
  $user_stmt->close();
} else {
  error_log("Error preparing user statement: " . $conn->error);
}

$admin_stmt->close();
} else {
  error_log("Error preparing admin statement: " . $conn->error);
}

$conn->close();
ob_end_flush(); // Flush output buffer and send output to the browser
?>
```

## 5. Database

### 1. dockerfile:

It used to php environment.  
It install MySQL extension to  
enable communication between  
php and the MySQL

```
Dockerfile > FROM
1  FROM php:apache
2
3  RUN docker-php-ext-install mysqli
4
5  # mysqli extension, which allows the php script to interact with a MySQL database.
```

2

3

## 2. Docker compose:

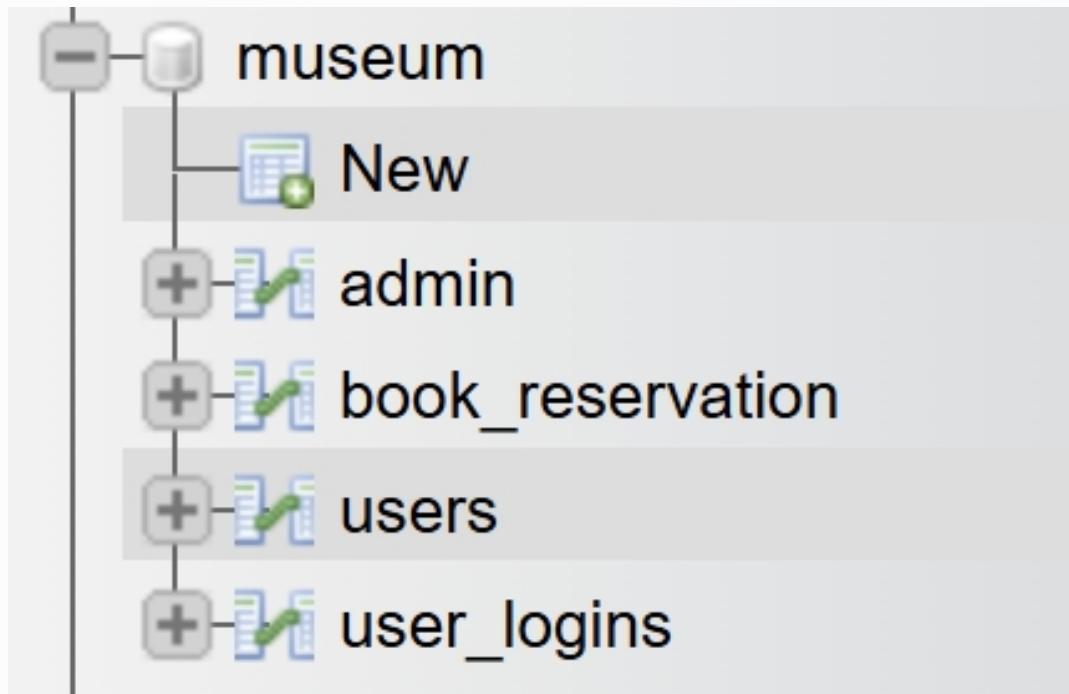
### Defined services for applications

```
docker-compose.yaml
1 version: '3.8'
2 # Setup with three Docker containers to provide a PHP environment with MySQL and phpMyAdmin
3
4 services:
5   mysql:
6     image: mysql:latest
7     container_name: mysql-wp-unique # Unique name for your MySQL container
8     environment:
9       MYSQL_ROOT_PASSWORD: root
10      MYSQL_DATABASE: museum # Change this to a unique database name
11      MYSQL_USER: Mohammadreza
12      MYSQL_PASSWORD: root
13     ports:
14       - "3307:3307" # Make sure this port doesn't conflict with other processes/containers
15     volumes:
16       - ./mysql_data:/var/lib/mysql # Ensure a unique volume path
17
18   php:
19     build:
20       context: .
21       dockerfile: Dockerfile
22     container_name: php-wp-unique # Unique name for your PHP container
23     ports:
24       - "8082:80" # Make sure this port doesn't conflict with other containers
25     volumes:
26       - ./src:/var/www/html
27     depends_on:
28       - mysql
29
30   phpmyadmin:
31     image: phpmyadmin:latest
32     container_name: phpmyadmin-wp-unique # Unique name for phpMyAdmin container
33     environment:
34       PMA_HOST: mysql
35       MYSQL_ROOT_PASSWORD: root
36     ports:
37       - "8081:80" # Make sure this port doesn't conflict with other containers
38     depends_on:
39       - mysql
```

- MySQL: run the MySQL database
- PHP : build php applications and run from port 8082
- Make web interface to manage MySQL database
- To build and start service i used command :  
Docker-compose up --build

And here is my data base tables

---



## 6. Conclusion:

---

Reza museum website offers an intuitive platform for both users and administrators. Users can efficiently manage their booking, while administrators maintain complete control over the system through robust management tools.

