Week 3 - Assignment

Anthony Meza

University of Arizona Global Campus

CST 499: Capstone for Computer Software Technology

Instructor Charmelia Butler

November 4, 2024

**Landing, Login, and Enrollment Pages Development**

To combine all the functions into a dynamic website that connects to the MySQL database, I started by setting up the foundational layout and navigation. The project was implemented using XAMPP, a popular tool for local development that provides Apache, MySQL, and PHP functionalities.

**Step 1: Layout Creation**

I began by creating a basic layout for the portal in index.php. This page serves as the homepage and includes navigation links to other pages such as the contact form, login, and registration pages. The initial structure of index.php ensured a clear path for users to navigate through the site.

**Step 2: Registration Functionality**

Next, I developed the registration functionality. The registration.php page provides a form for new users to submit their details. The form data is processed by register.php, which interacts with the MySQL database. In this step, I also designed the MySQL table (tblUser) to store user information. This includes fields for email, password, first name, last name, address, phone number, salary, and SSN.

**Step 3: Login Functionality**

For user authentication, I created a login page (login.php). Users enter their credentials, which are validated against the database in authenticate.php. Upon successful login, users are redirected to profile.php, where they can view their personal information. I also implemented session handling to control access based on the user's login status, which dynamically adjusts the navigation menu in index.php.

**Step 4: Profile and Logout**

The profile.php page displays the user's profile information after they have logged in. This page retrieves data from the database based on the session details. For logging out, I created logout.php to end the user session and redirect them back to the homepage.

**Step 5: Contact Us Page**

The contact page (contactus.php) that I began to implement allows users to send messages. I did not have enough time for the website to actually give feedback for the form data such as “message sent” so I just left it as is for show.

Purpose of the Employee Portal Website

The primary purpose of the employee portal website is to provide a centralized platform for employees to manage their personal information and communicate with the organization. The portal includes features such as user registration, login, profile management, and a contact form for inquiries or feedback. By consolidating these functionalities, the portal aims to streamline employee interactions and improve data management efficiency.

Home Page (index.php)

Description: The Home Page serves as the central hub of the employee portal. It welcomes users to the site and provides navigation links to other key sections, including the Contact Us page, Login, and Registration pages. Depending on the user’s login status, it dynamically displays different links. For logged-in users, it shows links to their Profile and a Logout option, while for those not logged in, it offers options to Login or Register.

<!DOCTYPE html>

<html>

<head>

<title>Employee Portal</title>

</head>

<body>

<h1>Welcome to the Employee Portal</h1>

<nav>

<ul>

<li><a href="index.php">Home</a></li>

<li><a href="contactus.php">Contact Us</a></li>

<li><a href="login.php">Login</a></li>

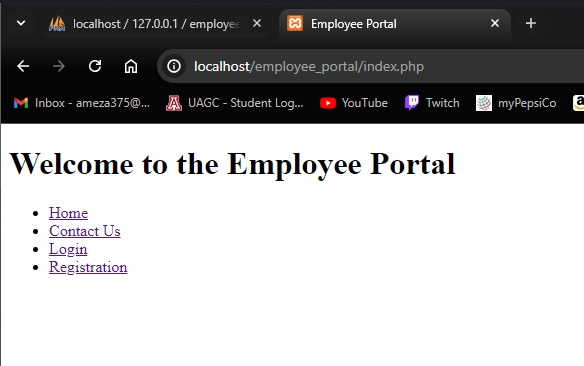
<li><a href="registration.php">Registration</a></li>

</ul>

</nav>

</body>

</html>



Registration Page (registration.php)

Description: The Registration Page allows new users to create an account in the employee portal. The form collects essential personal details such as email, password, first name, last name, address, phone number, salary, and Social Security Number (SSN). Upon submission, this data is sent to register.php, which processes and stores the information in the database, enabling the user to access the portal’s features after successful registration.

<!DOCTYPE html>

<html>

<head>

<title>Employee Registration</title>

</head>

<body>

<h1>Employee Registration</h1>

<form action="register.php" method="POST">

Email: <input type="email" name="email" required><br>

Password: <input type="password" name="password" required><br>

First Name: <input type="text" name="firstName" required><br>

Last Name: <input type="text" name="lastName" required><br>

Address: <input type="text" name="address" required><br>

Phone: <input type="text" name="phone" required><br>

Salary: <input type="number" name="salary" required><br>

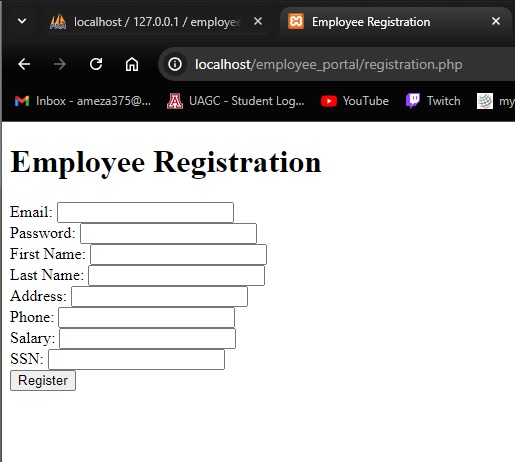
SSN: <input type="text" name="ssn" required><br>

<input type="submit" value="Register">

</form>

</body>

</html>



Registration Processing (register.php)

Description: This PHP script handles the data submitted from the Registration Page. It connects to the MySQL database, prepares an SQL statement to insert the new user’s data into the tblUser table, and executes the statement. The script also hashes the user’s password for security purposes before storing it. It provides feedback on whether the registration was successful or if an error occurred.

<?php

$servername = "localhost";

$username = "root";

$password = "";

$dbname = "employee\_portal";

// Check if the form was submitted

if ($\_SERVER["REQUEST\_METHOD"] == "POST") {

// Create connection

$conn = new mysqli($servername, $username, $password, $dbname);

// Check connection

if ($conn->connect\_error) {

die("Connection failed: " . $conn->connect\_error);

}

// Prepare and bind

$stmt = $conn->prepare("INSERT INTO tblUser (email, password, firstName, lastName, address, phone, salary, SSN) VALUES (?, ?, ?, ?, ?, ?, ?, ?)");

$stmt->bind\_param("ssssssis", $email, $hashed\_password, $firstName, $lastName, $address, $phone, $salary, $SSN);

// Collect and sanitize form inputs

$email = filter\_input(INPUT\_POST, 'email', FILTER\_SANITIZE\_EMAIL);

$hashed\_password = password\_hash($\_POST['password'], PASSWORD\_DEFAULT);

$firstName = filter\_input(INPUT\_POST, 'firstName', FILTER\_SANITIZE\_STRING);

$lastName = filter\_input(INPUT\_POST, 'lastName', FILTER\_SANITIZE\_STRING);

$address = filter\_input(INPUT\_POST, 'address', FILTER\_SANITIZE\_STRING);

$phone = filter\_input(INPUT\_POST, 'phone', FILTER\_SANITIZE\_STRING);

$salary = filter\_input(INPUT\_POST, 'salary', FILTER\_SANITIZE\_NUMBER\_FLOAT, FILTER\_FLAG\_ALLOW\_FRACTION);

if (!isset($\_POST['SSN'])) {

die("Error: SSN field is missing.");

}

$SSN = filter\_input(INPUT\_POST, 'SSN', FILTER\_SANITIZE\_STRING);

// Execute the statement

if ($stmt->execute()) {

echo "New record created successfully";

} else {

echo "Error: " . $stmt->error;

}

// Close the statement and connection

$stmt->close();

$conn->close();

}

?>

<!DOCTYPE html>

<html>

<head>

<title>Registration Form</title>

</head>

<body>

<h2>Registration Form</h2>

<form method="POST" action="register.php">

<label for="email">Email:</label><br>

<input type="email" id="email" name="email" required><br><br>

<label for="password">Password:</label><br>

<input type="password" id="password" name="password" required><br><br>

<label for="firstName">First Name:</label><br>

<input type="text" id="firstName" name="firstName" required><br><br>

<label for="lastName">Last Name:</label><br>

<input type="text" id="lastName" name="lastName" required><br><br>

<label for="address">Address:</label><br>

<input type="text" id="address" name="address" required><br><br>

<label for="phone">Phone:</label><br>

<input type="text" id="phone" name="phone" required><br><br>

<label for="salary">Salary:</label><br>

<input type="number" step="0.01" id="salary" name="salary" required><br><br>

<label for="SSN">SSN:</label><br>

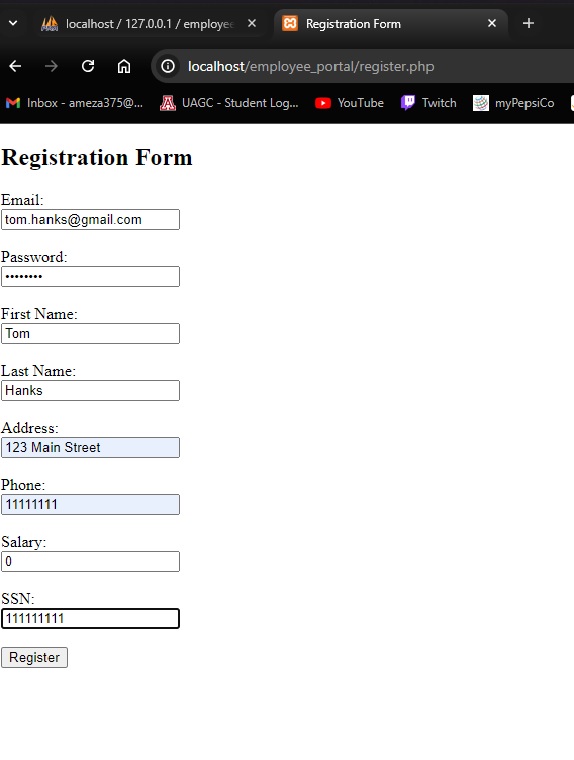
<input type="text" id="SSN" name="SSN" required><br><br>

<input type="submit" value="Register">

</form>

</body>

</html>



Login Page (login.php)

Description: The Login Page enables users to authenticate their credentials to access the employee portal. Users input their email and password, which are sent to authenticate.php for validation. This page is crucial for securing access to the portal and ensures that only authorized users can log in and view sensitive information.

<!DOCTYPE html>

<html>

<head>

<title>Employee Login</title>

</head>

<body>

<h1>Employee Login</h1>

<form action="authenticate.php" method="POST">

Email: <input type="email" name="email" required><br>

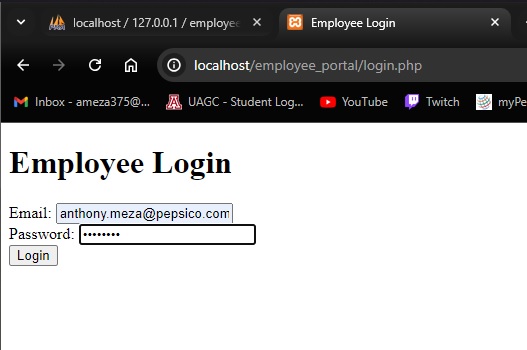
Password: <input type="password" name="password" required><br>

<input type="submit" value="Login">

</form>

</body>

</html>



Login Authentication (authenticate.php)

Description: The authenticate.php script processes the login information submitted from the Login Page. It queries the database to retrieve user data and verifies the password using password hashing techniques. If the credentials match, it initiates a session and redirects the user to their Profile Page. If authentication fails, it provides an error message.

<?php

session\_start();

$servername = "localhost";

$username = "root";

$password = "";

$dbname = "employee\_portal";

// Create connection

$conn = new mysqli($servername, $username, $password, $dbname);

// Check connection

if ($conn->connect\_error) {

die("Connection failed: " . $conn->connect\_error);

}

// Prepare and execute

$email = $\_POST['email'];

$password = $\_POST['password'];

$sql = "SELECT \* FROM tblUser WHERE email = ?";

$stmt = $conn->prepare($sql);

$stmt->bind\_param("s", $email);

$stmt->execute();

$result = $stmt->get\_result();

$user = $result->fetch\_assoc();

if ($user && password\_verify($password, $user['password'])) {

$\_SESSION['user\_id'] = $user['id'];

header("Location: profile.php");

} else {

echo "Invalid email or password";

}

$stmt->close();

$conn->close();

?>

Profile Page (profile.php)

Description: The Profile Page displays the personal information of a logged-in user, including email, first name, last name, address, phone number, salary, and SSN. It can only be accessed by users who are logged in, ensuring that personal data is protected and viewable only by the authorized user.

<?php

session\_start();

if (!isset($\_SESSION['user\_id'])) {

header("Location: login.php");

exit();

}

$servername = "localhost";

$username = "root";

$password = "";

$dbname = "employee\_portal";

// Create connection

$conn = new mysqli($servername, $username, $password, $dbname);

// Check connection

if ($conn->connect\_error) {

die("Connection failed: " . $conn->connect\_error);

}

$user\_id = $\_SESSION['user\_id'];

$sql = "SELECT \* FROM tblUser WHERE id = ?";

$stmt = $conn->prepare($sql);

$stmt->bind\_param("i", $user\_id);

$stmt->execute();

$result = $stmt->get\_result();

$user = $result->fetch\_assoc();

?>

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<title>Profile</title>

</head>

<body>

<h1>Profile</h1>

<p>Email: <?php echo $user['email']; ?></p>

<p>First Name: <?php echo $user['firstName']; ?></p>

<p>Last Name: <?php echo $user['lastName']; ?></p>

<p>Address: <?php echo $user['address']; ?></p>

<p>Phone: <?php echo $user['phone']; ?></p>

<p>Salary: <?php echo $user['salary']; ?></p>

<p>SSN: <?php echo $user['SSN']; ?></p>

<a href="logout.php">Logout</a>

</body>

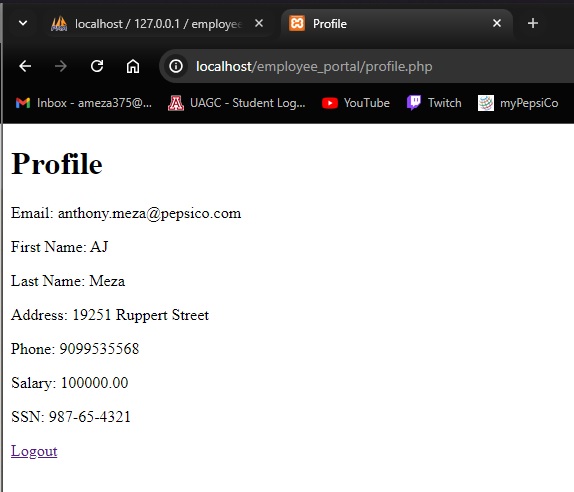
</html>

<?php

$stmt->close();

$conn->close();

?>



Logout Script (logout.php)

Description: The Logout Script is used to end the current user session, effectively logging the user out of the portal. It clears session data and redirects the user back to the Home Page. This functionality is essential for maintaining security and privacy, allowing users to safely exit their accounts.

<?php

session\_start();

session\_unset();

session\_destroy();

header("Location: index.php");

?>