# Assignment 6

**Student Name:** Himanshu Gupta **UID:** 23BCS10889

**Branch:** BE CSE **Section/Group:** Web Program.

**Semester:** 5th

1. **Aim:-**  **To Design and Implement a Complete Social Site**

# CODE:

# Dashboard.php

<?php

// 1) Enable PHP error reporting error\_reporting(E\_ALL); ini\_set('display\_errors', 1);

// 2) Start session and connect to the database session\_start();

$conn = new mysqli("localhost", "root", "", "user\_system"); if ($conn->connect\_error) {

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

}

// 3) (For testing) Simulate a logged-in user if none exists if (!isset($\_SESSION['email\_or\_phone'])) {

$\_SESSION['email\_or\_phone'] = ['test@example.com';](mailto:%27test@example.com)

}

$email = $\_SESSION['email\_or\_phone'];

// 4) Fetch the user’s profile picture

$stmt = $conn->prepare("SELECT profile\_pic FROM users WHERE email\_or\_phone = ?");

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

$stmt->execute();

$user = $stmt->get\_result()->fetch\_assoc();

$profilePic = $user['profile\_pic'] ?? 'uploads/default-profile.png';

// 5) Fetch the count of pending friend requests

$noti\_stmt = $conn->prepare("SELECT COUNT(\*) AS total FROM friend\_requests WHERE receiver\_email = ? AND status = 'pending'");

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

$noti\_stmt->execute();

$pending\_count = $noti\_stmt->get\_result()->fetch\_assoc()['total'];

?>

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<title>Dashboard</title>

<style>

/\* Layout \*/

body { display: flex; margin: 0; font-family: Segoe UI; background: #f4f6f8; }

.sidebar {

width: 240px; background: #2c3e50; color: white; display: flex; flex-direction: column; align-items: center; padding-top: 30px; height: 100vh;

}

.sidebar img {

width: 100px; height: 100px; border-radius: 50%;

border: 3px solid white; object-fit: cover; margin-bottom: 10px;

}

.sidebar h3 { margin-bottom: 20px; text-align: center; word-break: break-word; }

.sidebar a {

color: white; text-decoration: none; padding: 12px 20px; width: 100%; display: block; text-align: left; cursor: pointer;

}

.sidebar a:hover { background: #1abc9c; }

.main { flex: 1; padding: 30px; overflow-y: auto; }

.section { display: none; }

.section.active { display: block; animation: fadeIn 0.3s ease-in-out; }

/\* Friend box \*/

.friend-box {

display: flex; align-items: center; background: white; padding: 10px; margin-bottom: 10px; border-radius: 6px;

}

.friend-box img { width: 40px; height: 40px; border-radius: 50%; margin-right: 10px; }

.friend-box span { flex: 1; }

.friend-box button { margin-left: 5px; padding: 5px 10px; border-radius: 5px; }

.friend-box .reject { background: red; color: white; }

{

background: white; height: 300px; overflow-y: auto; padding: 10px; border: 1px solid #ccc; border-radius: 8px;

}

@keyframes fadeIn {

from { opacity: 0; transform: translateY(5px); } to { opacity: 1; transform: translateY(0); }

}

</style>

</head>

<body>

<!-- Sidebar -->

<div class="sidebar">

<img src="<?php echo htmlspecialchars($profilePic); ?>" alt="Profile">

<h3><?php echo htmlspecialchars($email); ?></h3>

<a data-section="home">Home</a>

<a data-section="friends">Friends</a>

<a data-section="messages">Messages</a>

<a data-section="notifications">Notifications <strong id="noti-badge"><?php echo

$pending\_count > 0 ? "({$pending\_count})" : ""; ?></strong></a>

<a data-section="scrap">Scrap</a>

<a data-section="post">Post</a>

<a data-section="setting">Setting</a>

<a data-section="game">Game</a>

<a data-section="groupchat">Group Chat</a>

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

</div>

<div class="main">

<!-- Home -->

<div id="home" class="section active">

<h2>Welcome, <?php echo htmlspecialchars($email); ?>!</h2>

<button onclick="showSection('upload')"> Update Profile Picture</button>

</div>

<!-- Upload -->

<div id="upload" class="section">

<h3> Upload Profile Picture</h3>

<form action="upload.php" method="POST" enctype="multipart/form-data">

<input type="file" name="profile\_pic" required><br><br>

<input type="hidden" name="email\_or\_phone" value="<?php echo htmlspecialchars($email);

?>">

<button type="submit">Upload</button>

</form>

</div>

<!-- Friends -->

<div id="friends" class="section">

<h3> Search Friends</h3>

<form action="search\_friend.php" method="POST">

<input type="text" name="search\_query" required>

<button type="submit">Search</button>

</form>

<hr>

<h3> Friend Requests</h3>

<?php

$rq = $conn->prepare("SELECT sender\_email FROM friend\_requests WHERE receiver\_email=? AND status='pending'");

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

$rq->execute();

$res = $rq->get\_result(); if ($res->num\_rows) {

while ($r = $res->fetch\_assoc()) {

$s = htmlspecialchars($r['sender\_email']); echo "<div class='friend-box'>

<img src='uploads/default-profile.png'><span>$s</span>

<form style='display:inline' action='handle\_request.php' method='POST'>

<input type='hidden' name='sender\_email' value='$s'>

<input type='hidden' name='action' value='accept'>

<button>Accept</button>

</form>

<form style='display:inline' action='handle\_request.php' method='POST'>

<input type='hidden' name='sender\_email' value='$s'>

<input type='hidden' name='action' value='reject'>

<button class='reject'>Reject</button>

</form>

</div>";

}

} else {

echo "<p>No pending friend requests.</p>";

}

?>

<hr>

<h3> My Friends</h3>

<?php

$fl = $conn->prepare(

"SELECT CASE WHEN sender\_email=? THEN receiver\_email ELSE sender\_email END AS friend\_email

FROM friend\_requests

WHERE (sender\_email=? OR receiver\_email=?) AND status='accepted'"

);

$fl->bind\_param("sss", $email, $email, $email);

$fl->execute();

$fs = $fl->get\_result();

while ($f = $fs->fetch\_assoc()) {

$fe = htmlspecialchars($f['friend\_email']); echo "<div class='friend-box'>

<img src='uploads/default-profile.png'><span>$fe</span>

<form action='unfriend.php' method='POST'>

<input type='hidden' name='friend\_email' value='$fe'>

<button class='reject'>Unfriend</button>

</form>

</div>";

}

?>

</div>

<!-- Messages -->

<div id="messages" class="section">

<h3> Chat with Friend</h3>

<select id="friend-selector" onchange="loadMessages(this.value)">

<option disabled selected>Choose a friend</option>

<?php

$fl->execute();

$fs = $fl->get\_result();

while ($f = $fs->fetch\_assoc()) {

$fe = htmlspecialchars($f['friend\_email']); echo "<option value='$fe'>$fe</option>";

}

?>

</select>

<div id="chat-box"></div>

</div>

<!-- Notifications -->

<div id="notifications" class="section">

<h3> Friend Request Notifications</h3>

<?php

$nq = $conn->prepare( "SELECT sender\_email, sent\_at FROM friend\_requests

WHERE receiver\_email=? AND status='pending' ORDER BY sent\_at DESC"

);

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

$nq->execute();

$noti = $nq->get\_result(); if ($noti->num\_rows) {

while ($n = $noti->fetch\_assoc()) {

$s = htmlspecialchars($n['sender\_email']);

$t = date("d M Y, h:i A", strtotime($n['sent\_at']));

echo "<p> <strong>$s</strong> sent you a request <small>($t)</small></p>";

}

} else {

echo "<p>No new notifications.</p>";

}

?>

</div>

<!-- Placeholder Sections -->

<div id="scrap" class="section"><h3> Scrap - Coming Soon</h3></div>

<div id="post" class="section"><h3> Post - Coming Soon</h3></div>

<div id="setting" class="section"><h3> Settings - Coming Soon</h3></div>

<div id="game" class="section"><h3> Game - Coming Soon</h3></div>

<div id="groupchat" class="section"><h3> Group Chat - Coming Soon</h3></div>

</div>

<!-- JavaScript -->

<script>

// Test JS load alert("JS is working!");

// Show a given section function showSection(id) {

document.querySelectorAll('.section').forEach(s => s.classList.remove('active')); document.getElementById(id).classList.add('active');

}

// Sidebar navigation

document.querySelectorAll('.sidebar a[data-section]').forEach(a => { a.addEventListener('click', e => {

e.preventDefault(); showSection(a.getAttribute('data-section'));

});

});

// Update notification badge periodically function updateNotificationBadge() { fetch('get\_notification\_count.php')

.then(r => r.text())

.then(t => {

document.getElementById('noti-badge').textContent = parseInt(t) > 0 ? `(${t})` : '';

});

}

updateNotificationBadge(); setInterval(updateNotificationBadge, 10000);

// Load chat messages function loadMessages(friend) {

const box = document.getElementById('chat-box'); box.innerHTML = '<em>Loading...</em>';

fetch('fetch\_messages.php?receiver=' + encodeURIComponent(friend))

.then(r => r.json())

.then(data => { box.innerHTML = ''; data.forEach(m => {

const d = document.createElement('div');

d.textContent = `${m.sender\_email === <?php echo json\_encode($email); ?> ? 'You' : m.sender\_email}: ${m.message}`;

box.appendChild(d);

});

});

}

</script>

</body>

</html>

# fetch\_messages.php

<?php session\_start();

header('Content-Type: application/json');

$conn = new mysqli("localhost", "root", "", "user\_system");

if (!isset($\_SESSION['email\_or\_phone']) || !isset($\_GET['receiver'])) { echo json\_encode([]);

exit;

}

$sender = $\_SESSION['email\_or\_phone'];

$receiver = $\_GET['receiver'];

$stmt = $conn->prepare("SELECT sender\_email, message FROM messages WHERE (sender\_email = ? AND receiver\_email = ?)

OR (sender\_email = ? AND receiver\_email = ?) ORDER BY sent\_at ASC");

$stmt->bind\_param("ssss", $sender, $receiver, $receiver, $sender);

$stmt->execute();

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

$messages = [];

while ($row = $result->fetch\_assoc()) {

$messages[] = $row;

}

echo json\_encode($messages);

# send\_message.php

<?php session\_start();

$conn = new mysqli("localhost", "root", "", "user\_system");

if (!isset($\_SESSION['email\_or\_phone'], $\_POST['receiver\_email'], $\_POST['message'])) { exit("Invalid input");

}

$sender = $\_SESSION['email\_or\_phone'];

$receiver = $\_POST['receiver\_email'];

$message = trim($\_POST['message']);

if ($message !== '') {

$stmt = $conn->prepare("INSERT INTO messages (sender\_email, receiver\_email, message) VALUES (?, ?, ?)");

$stmt->bind\_param("sss", $sender, $receiver, $message);

$stmt->execute();

}

header("Location: dashboard.php"); exit;

# search\_friend.php

i\_set('display\_errors', 1); session\_start();

$conn = new mysqli("localhost", "root", "", "user\_system"); if ($conn->connect\_error) {

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

}

$currentUser = $\_SESSION['email\_or\_phone'];

$search = $\_POST['search\_query'] ?? '';

$sql = "SELECT email\_or\_phone FROM users WHERE email\_or\_phone LIKE ?

AND email\_or\_phone != ?";

$searchTerm = '%' . $search . '%';

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

$stmt->bind\_param("ss", $searchTerm, $currentUser);

$stmt->execute();

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

?>

<!DOCTYPE html>

<html>

<head>

<title>Friend Search Results</title>

<style>body{

font-family: Arial; padding: 20px;

}

.user {

background: #f9f9f9; padding: 12px; margin-bottom: 10px; border-radius: 6px;

}

button {

background: #28a745; color: white;

padding: 6px 10px; border: none; border-radius: 4px; cursor: pointer;

}

</style>

</head>

<body>

<h2> Results for "<?php echo htmlspecialchars($search); ?>"</h2>

<a href="dashboard.php">⇦ Back to Dashboard</a><br><br>

<?php

if ($result->num\_rows > 0) {

while ($row = $result->fetch\_assoc()) {

$email = htmlspecialchars($row['email\_or\_phone']);

// Check if a friend request already exists

$check = $conn->prepare("SELECT status FROM friend\_requests WHERE sender\_email = ? AND receiver\_email = ?");

$check->bind\_param("ss", $currentUser, $row['email\_or\_phone']);

$check->execute();

$check\_result = $check->get\_result();

echo "<div class='user'><strong>$email</strong>"; if ($check\_result->num\_rows > 0) {

$req = $check\_result->fetch\_assoc();

$status = ucfirst($req['status']);

echo "<span class='status'>($status)</span>";

} else {

echo "<form method='POST' action='send\_request.php' style='display:inline; margin-left: 10px;'>

<input type='hidden' name='receiver\_email' value='$email'>

<button type='submit'>Send Friend Request</button>

</form>";

}

echo "</div>";

}

} else {

echo "<p> No users found.</p>";

}

?>

</body>

</html>

# send\_request.php

<?php session\_start();

$conn = new mysqli("localhost", "root", "", "user\_system"); if ($conn->connect\_error) {

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

}

$sender = $\_SESSION['email\_or\_phone'];

$receiver = $\_POST['receiver\_email'];

// Check if request already exists

$check = $conn->prepare("SELECT \* FROM friend\_requests WHERE sender\_email = ? AND receiver\_email = ?");

$check->bind\_param("ss", $sender, $receiver);

$check->execute();

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

if ($result->num\_rows > 0) {

echo "  You’ve already sent a request to this user.";

} else {

// Insert new request

$insert = $conn->prepare("INSERT INTO friend\_requests (sender\_email, receiver\_email, status) VALUES (?, ?, 'pending')");

$insert->bind\_param("ss", $sender, $receiver);

if ($insert->execute()) {

echo " Friend request sent to $receiver!";

} else {

echo " Failed to send request.";

}

}

?>

<br><br>

<a href='dashboard.php'>Back to Dashboard</a>

# handle\_request.php

<?php session\_start();

$conn = new mysqli("localhost", "root", "", "user\_system"); if ($conn->connect\_error) {

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

}

$receiver = $\_SESSION['email\_or\_phone'];

$sender = $\_POST['sender\_email'];

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

if ($action === 'accept') {

$status = 'accepted';

} else {

$status = 'rejected'; // Or you can use 'blocked' if you prefer

}

$update = $conn->prepare("UPDATE friend\_requests SET status = ? WHERE sender\_email = ? AND receiver\_email = ?");

$update->bind\_param("sss", $status, $sender, $receiver); if ($update->execute()) {

echo " Friend request has been $status.";

} else {

echo " Failed to update request.";

}

echo "<br><br><a href='dashboard.php'>Back to Dashboard</a>";

?>

# unfriend.php

<?php session\_start();

$conn = new mysqli("localhost", "root", "", "user\_system"); if ($conn->connect\_error) {

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

}

$current\_user = $\_SESSION['email\_or\_phone'];

$friend\_email = $\_POST['friend\_email'] ?? '';

if ($friend\_email) {

$stmt = $conn->prepare(" DELETE FROM friend\_requests

WHERE ((sender\_email = ? AND receiver\_email = ?) OR (sender\_email = ? AND receiver\_email = ?)) AND status = 'accepted'

");

$stmt->bind\_param("ssss", $current\_user, $friend\_email, $friend\_email, $current\_user);

if ($stmt->execute()) { header("Location: dashboard.php"); exit();

} else {

echo " Failed to unfriend.";

}

} else {

echo " Invalid request.";

}

?>

# register.html

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<title>Register</title><style> body{

background: #f1f1f1;

font-family: Arial, sans-serif; display: flex;

justify-content: center; align-items: center; height: 100vh;

margin: 0;

}

.container { background: #fff; padding: 30px;

border-radius: 10px;

box-shadow: 0 0 15px rgba(0,0,0,0.1); width: 350px;

}

h2 {

text-align: center; color: #333;

}

input, button { width: 100%; padding: 12px; margin: 10px 0; border-radius: 5px;

border: 1px solid #ccc;

}

button {

background-color: #007bff; color: white;

font-size: 16px; border: none;

}

button:hover {

background-color: #0069d9;

}

</style>

</head>

<body>

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

<h2>Register</h2>

<input type="text" name="email\_or\_phone" placeholder="Email or Phone" required>

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

<input type="password" name="confirm\_password" placeholder="Confirm Password" required>

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

<p style="text-align:center; margin-top: 15px;"> Already have an account?

<a href="login.html">Login here</a>

</p>

</form>

</body>

</html>

# register.php

<?php error\_reporting(E\_ALL); ini\_set('display\_errors', 1);

$conn = new mysqli("localhost", "root", "", "user\_system"); if ($conn->connect\_error) {

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

}

$email\_or\_phone = trim($\_POST['email\_or\_phone']);

$password = trim($\_POST['password']);

// Check if user already exists

$check = $conn->prepare("SELECT id FROM users WHERE email\_or\_phone = ?");

$check->bind\_param("s", $email\_or\_phone);

$check->execute();

$check->store\_result();

if ($check->num\_rows > 0) {

echo " User already registered! <a href='login.html'>Login here</a>"; exit;

}

// Hash the password

$hashedPassword = password\_hash($password, PASSWORD\_DEFAULT);

// Insert new user

$stmt = $conn->prepare("INSERT INTO users (email\_or\_phone, password, login\_attempts, status, profile\_pic) VALUES (?, ?, 0, 'active', 'uploads/default-profile.png')");

$stmt->bind\_param("ss", $email\_or\_phone, $hashedPassword);

if ($stmt->execute()) {

echo " Registration successful! <a href='login.html'>Login now</a>";

} else {

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

}

?>

# login.html

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<title>Login</title><style> body{

background: #f4f4f4;

font-family: Arial, sans-serif; display: flex;

justify-content: center; align-items: center; height: 100vh;

margin: 0;

}

.container { background: white; padding: 30px; border-radius: 10px;

box-shadow: 0 0 10px rgba(0,0,0,0.1); width: 350px;

}

input, button { width: 100%; padding: 12px;

margin: 10px 0; border-radius: 5px;

border: 1px solid #ccc;

}

button {

background: #28a745; color: white;

border: none;

}

button:hover { background: #218838;

}

</style>

</head>

<body>

<div class="container">

<h2>Login</h2>

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

<input type="text" name="email\_or\_phone" placeholder="Email or Phone" required>

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

<button type="submit">Login</button>

</form>

</div>

</body>

</html>

# login.php

<?php session\_start();

error\_reporting(E\_ALL); ini\_set('display\_errors', 1);

// Connect to DB

$conn = new mysqli("localhost", "root", "", "user\_system"); if ($conn->connect\_error) {

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

}

// Get user input

$email\_or\_phone = trim($\_POST['email\_or\_phone'] ?? '');

$password\_input = trim($\_POST['password'] ?? '');

// Lookup user

$stmt = $conn->prepare("SELECT \* FROM users WHERE email\_or\_phone = ?");

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

$stmt->execute();

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

if ($result->num\_rows === 1) {

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

// Check if account is locked

if ($user['status'] === 'locked') {

echo " Account is locked due to too many failed attempts."; exit;

}

//  Verify hashed password

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

// Reset login attempts

$reset = $conn->prepare("UPDATE users SET login\_attempts = 0 WHERE email\_or\_phone

= ?");

$reset->bind\_param("s", $email\_or\_phone);

$reset->execute();

$\_SESSION['email\_or\_phone'] = $email\_or\_phone; header("Location: dashboard.php");

exit();

} else {

// Incorrect password: increase attempts

$attempts = $user['login\_attempts'] + 1; if ($attempts >= 3) {

$lock = $conn->prepare("UPDATE users SET login\_attempts = ?, status = 'locked' WHERE email\_or\_phone = ?");

} else {

$lock = $conn->prepare("UPDATE users SET login\_attempts = ? WHERE email\_or\_phone

= ?");

}

$lock->bind\_param("is", $attempts, $email\_or\_phone);

$lock->execute();

echo " Incorrect password. Attempt $attempts of 3.";

}

} else {

echo " No user found with that email or phone.";

}

?>

# upload.php

<?php error\_reporting(E\_ALL); ini\_set('display\_errors', 1); session\_start();

// DB connection

$conn = new mysqli("localhost", "root", "", "user\_system"); if ($conn->connect\_error) {

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

}

// Get user email from POST

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

$newFile = $\_FILES['profile\_pic'];

// Upload path

$target\_dir = "uploads/";

$filename = time() . "\_" . basename($newFile["name"]);

$target\_file = $target\_dir . $filename;

// Get current profile picture

$sql = "SELECT profile\_pic FROM users WHERE email\_or\_phone = ?";

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

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

$stmt->execute();

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

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

$old\_pic = $user['profile\_pic'];

// Upload new image

if (move\_uploaded\_file($newFile["tmp\_name"], $target\_file)) {

// Delete old image if not default

if ($old\_pic !== 'uploads/default-profile.png' && file\_exists($old\_pic)) { unlink($old\_pic);

}

// Update DB

$update = $conn->prepare("UPDATE users SET profile\_pic = ? WHERE email\_or\_phone = ?");

$update->bind\_param("ss", $target\_file, $email);

if ($update->execute()) {

echo " Profile picture updated successfully!<br>"; echo "<a href='dashboard.php'>Back to Dashboard</a>";

} else {

echo " Failed to update database.";

}

} else {

echo " Failed to upload file.";

}

?>

# dashboard.php

<?php

// 1) Enable error reporting error\_reporting(E\_ALL); ini\_set('display\_errors', 1);

// 2) Start session and connect session\_start();

$conn = new mysqli("localhost", "root", "", "user\_system"); if ($conn->connect\_error) {

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

}

// 3) Ensure a logged-in user (for testing—remove when real login is in place) if (!isset($\_SESSION['email\_or\_phone'])) {

$\_SESSION['email\_or\_phone'] = ['test@example.com';](mailto:%27test@example.com)

}

$email = $\_SESSION['email\_or\_phone'];

// 4) Fetch profile picture

$stmt = $conn->prepare("SELECT profile\_pic FROM users WHERE email\_or\_phone = ?");

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

$stmt->execute();

$user = $stmt->get\_result()->fetch\_assoc();

$profilePic = $user['profile\_pic'] ?? 'uploads/default-profile.png';

// 5) Fetch pending notification count

$noti\_stmt = $conn->prepare(

"SELECT COUNT(\*) AS total FROM friend\_requests WHERE receiver\_email = ? AND status = 'pending'"

);

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

$noti\_stmt->execute();

$pending\_count = $noti\_stmt->get\_result()->fetch\_assoc()['total'];

?>

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<title>Dashboard</title>

<style>

body { display:flex; margin:0; font-family:Segoe UI; background:#f4f6f8; }

.sidebar {

width:220px; background:#2c3e50; color:white; display:flex; flex-direction:column; align-items:center; padding-top:30px; height:100vh;

}

.sidebar img {

width:100px; height:100px; border-radius:50%;

border:3px solid white; object-fit:cover; margin-bottom:10px;

}

.sidebar h3 { margin-bottom:20px; text-align:center; word-break:break-word; }

.sidebar a {

color:white; text-decoration:none; padding:12px 20px; width:100%; display:block; text-align:left; cursor:pointer;

}

.sidebar a:hover { background:#1abc9c; }

.main { flex:1; padding:30px; overflow-y:auto; }

.section { display:none; }

.section.active { display:block; animation:fadeIn .3s ease-in-out; } @keyframes fadeIn {

from { opacity:0; transform:translateY(5px); } to { opacity:1; transform:translateY(0); }

}

</style>

</head>

<body>

<div class="sidebar">

<img src="<?php echo htmlspecialchars($profilePic); ?>" alt="Profile">

<h3><?php echo htmlspecialchars($email); ?></h3>

<a data-section="home">Home</a>

<a data-section="notifications">

Notifications <strong id="noti-badge"><?php echo $pending\_count>0?"(".

$pending\_count.")":""; ?></strong>

</a>

<a data-section="scrap">Scrap</a>

<a data-section="post">Post</a>

<a data-section="setting">Setting</a>

<a data-section="game">Game</a>

<a data-section="groupchat">Group Chat</a>

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

</div>

<div class="main">

<div id="home" class="section active">

<h2>Welcome, <?php echo htmlspecialchars($email); ?>!</h2>

<p>This is your dashboard home.</p>

</div>

<div id="notifications" class="section">

<h2> Friend Request Notifications</h2>

<?php

$nq = $conn->prepare( "SELECT sender\_email, sent\_at FROM friend\_requests

WHERE receiver\_email = ? AND status = 'pending' ORDER BY sent\_at DESC"

);

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

$nq->execute();

$notes = $nq->get\_result(); if ($notes->num\_rows) {

while ($n = $notes->fetch\_assoc()) {

$s = htmlspecialchars($n['sender\_email']);

$t = date("d M Y, h:i A", strtotime($n['sent\_at']));

echo "<p> <strong>{$s}</strong> sent you a request <small>($t)</small></p>";

}

} else {

echo "<p>No new notifications.</p>";

}

?>

</div>

<div id="scrap" class="section">

<h2> Scrap</h2>

<p>Coming soon…</p>

</div>

<div id="post" class="section">

<h2> Post</h2>

<p>Coming soon…</p>

</div>

<div id="setting" class="section">

<h2> Settings</h2>

<p>Coming soon…</p>

</div>

<div id="game" class="section">

<h2> Game</h2>

<p>Coming soon…</p>

</div>

<div id="groupchat" class="section">

<h2> Group Chat</h2>

<p>Coming soon…</p>

</div>

</div>

<script>

// Section toggling

document.querySelectorAll('.sidebar a[data-section]').forEach(link => {

link.addEventListener('click', e => { e.preventDefault();

const sec = link.getAttribute('data-section'); document.querySelectorAll('.section').forEach(s => s.classList.remove('active')); document.getElementById(sec).classList.add('active');

});

});

// Update notification badge function updateBadge() { fetch('get\_notification\_count.php')

.then(r => r.text())

.then(c => {

document.getElementById('noti-badge').textContent = parseInt(c) > 0 ? `(${c})` : '';

});

}

updateBadge(); setInterval(updateBadge, 10000);

</script>

</body>

</html>

# sql code

users Table (for login, registration, profile picture)

CREATE TABLE users (

id INT AUTO\_INCREMENT PRIMARY KEY, name VARCHAR(100) NOT NULL,

email VARCHAR(100) UNIQUE NOT NULL, password VARCHAR(255) NOT NULL,

profile\_picture VARCHAR(255) DEFAULT 'default.jpg'

);

friend\_requests Table

CREATE TABLE friend\_requests (

id INT AUTO\_INCREMENT PRIMARY KEY,

sender\_id INT NOT NULL, receiver\_id INT NOT NULL,

status ENUM('pending', 'accepted', 'rejected', 'blocked') DEFAULT 'pending', created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

FOREIGN KEY (sender\_id) REFERENCES users(id) ON DELETE CASCADE,

FOREIGN KEY (receiver\_id) REFERENCES users(id) ON DELETE CASCADE

);

messages Table (for private chats - basic version)

CREATE TABLE messages (

id INT AUTO\_INCREMENT PRIMARY KEY,

sender\_id INT NOT NULL, receiver\_id INT NOT NULL, message TEXT NOT NULL,

sent\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

FOREIGN KEY (sender\_id) REFERENCES users(id) ON DELETE CASCADE,

FOREIGN KEY (receiver\_id) REFERENCES users(id) ON DELETE CASCADE

);

notifications Table (for showing friend request/message alerts)

CREATE TABLE notifications (

id INT AUTO\_INCREMENT PRIMARY KEY,

user\_id INT NOT NULL, message TEXT NOT NULL,

is\_read BOOLEAN DEFAULT FALSE,

created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

FOREIGN KEY (user\_id) REFERENCES users(id) ON DELETE CASCADE

);

# OUTPUT:-

# 

# 

# 

# 

1. **LEARNING OUTCOMES:-**

* **Form Creation and User Input Handling**
* **Responsive UI Design with HTML & CSS**
* **Client-Server Communication Using AJAX**
* **Backend Development with PHP**
* **Database Design and CRUD Operations**
* **Session Management and User Authentication**
* **Real-Time Data Exchange and Dynamic Updates**
* **Status Handling in User Interactions**
* **Security Measures and Input Validation**
* **Error Handling and User Feedback**
* **Improved User Experience Through Interactivity**
* **Integration of Multiple Components into Functional Systems**