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

2. 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';
}
   $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_row
    s) {
      while ($r = $res->fetch_assoc()) {
        $s =
        htmlspecialchars($r['sender_e
        mail']); 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.innerH
```

```
          TML = '';
          data.forEac
          h(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('dis
     play_err
     ors', 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 {
  backgrou
  nd:
  #f9f9f9;
  padding:
  12px;
  margin-
  bottom:
  10px;
  border-
  radius:
  6px;
 }
 button {
  background:
  #28a745;
  color:
  white;
  paddin
  g: 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 {
      backgroun
      d: #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_reportin
    g(E_ALL);
    ini_set('displa
    y_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_reportin
    g(E_ALL);
    ini_set('displa
    y_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_reportin
g(E_ALL);
ini_set('displa
y_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';
  }
      $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_co
     unt.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
);
```

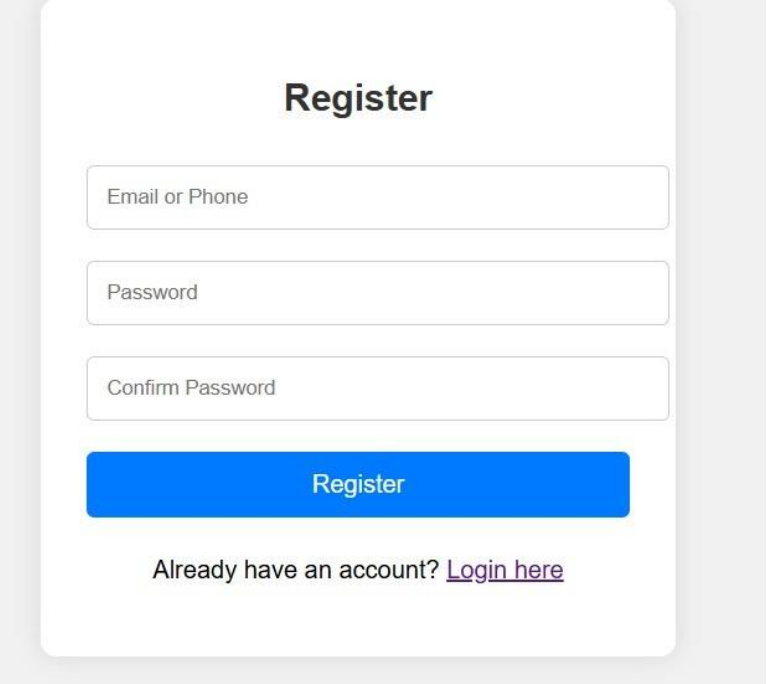
✅ messages Table (for private chats - basic version)

```
   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
);
```

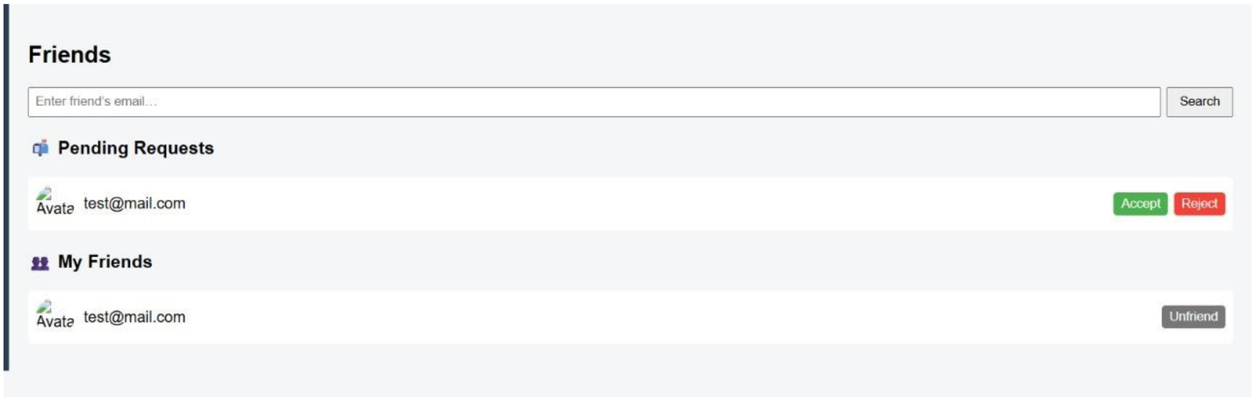
notifications Table (for showing friend request/message alerts)

```
  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
);
```

3. OUTPUT:-

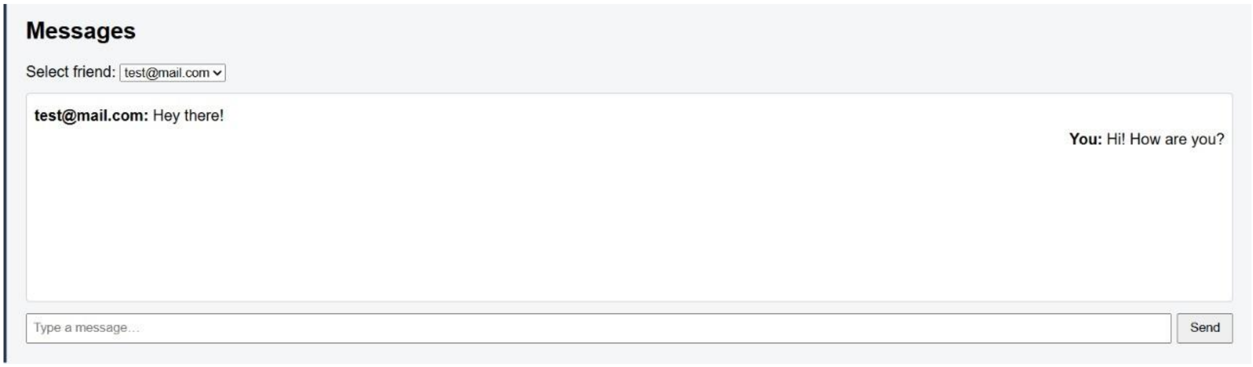


Already have an account? Login here



test@mail.com

test@mail.com

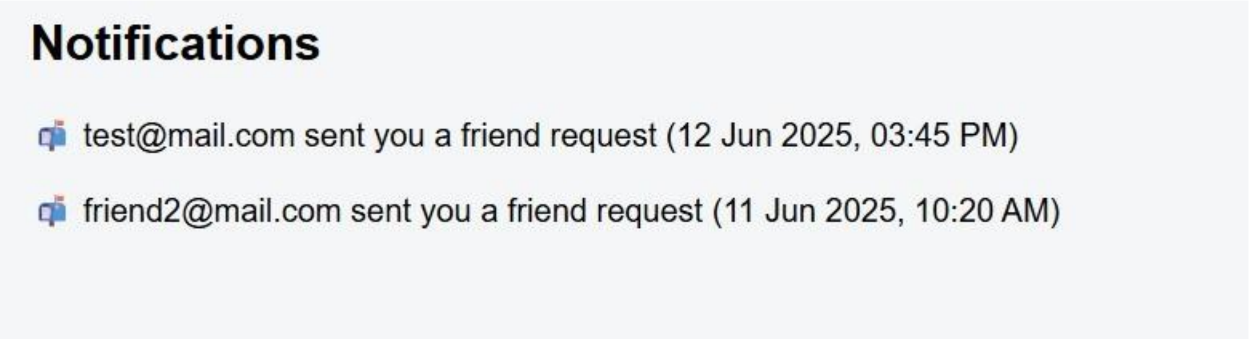


Messages

Select friend: test@mail.com

test@mail.com: Hey there!

You: Hi! How are you?



Notifications

📬 test@mail.com sent you a friend request (12 Jun 2025, 03:45 PM)

📬 friend2@mail.com sent you a friend request (11 Jun 2025, 10:20 AM)

4. 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