

PROJECT PROPOSAL

TALENTFORGE



MOHOMED HOOSSEN : 402103099

KALDEN PILLAY : 402100568

YODIN MAISTRY : 402104627

SHERISAN DEAN : 402100151

AARON SINGH : 402104931

Contents

Table of Contents

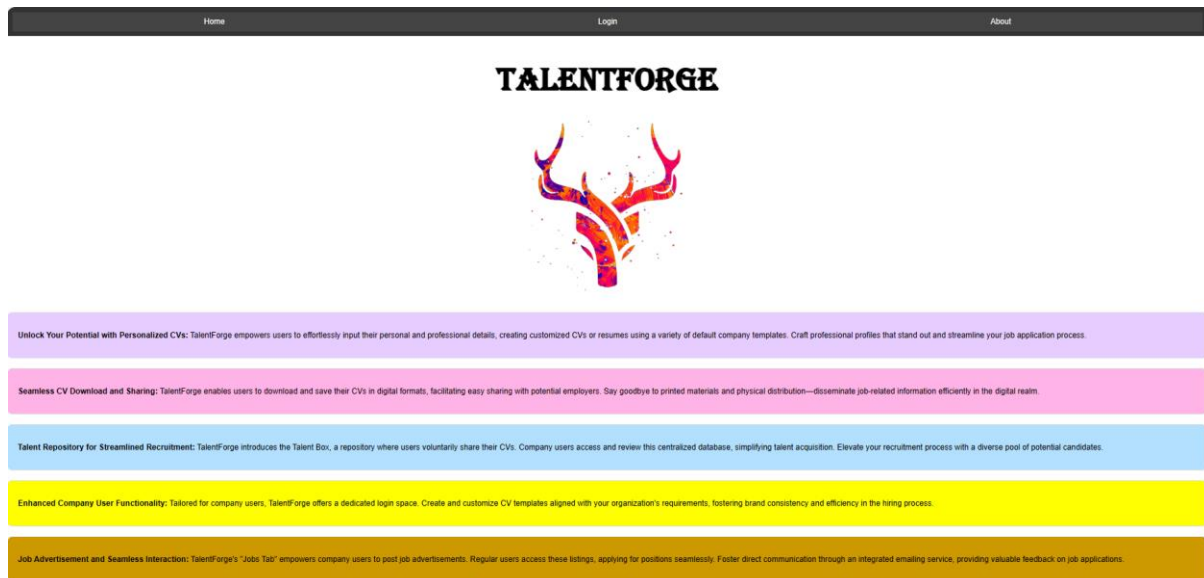
Coding 3

Testing 77

TESTCASE:..... 90

Coding

TalentForgeHomePage.php:



```
<!DOCTYPE html>
<html>
<head>
  <title>TalentForge</title>
  <style>
    body {
      font-family: Arial, sans-serif;
      text-align: center;
      margin: 0px;
    }

    h1 {
      font-size: 66px;
      font-weight: bold;
      font-family: 'ALGERIAN', sans-serif;
      margin-bottom: 0px;
    }

    #logo {
      width: 430px; /* Adjust the width as needed */
      height: auto;
    }

    .tile:nth-child(1) {
      background-color: #e6ccff; /* Light Purple */
    }

    .tile:nth-child(2) {
      background-color: #ffb3e6; /* Pink */
    }

    .tile:nth-child(3) {
```

```

        background-color: #b3e0ff; /* Light Blue */
    }

    .tile:nth-child(4) {

        background-color: #ffff00; /* Orange */
    }

    .tile:nth-child(5) {
        background-color: #cc9900; /* Dark Yellow */
    }

    .tile:nth-child(6) {
        background-color: #ffddb3; /* Light Orange */
    }

    .tile {
        border: 1px solid #ddd;
        border-radius: 10px;
        padding: 20px;
        margin-bottom: 20px;
        text-align: left;
    }

table label {
    font-size: 24px; /* Adjust the font size as needed */
}

table button {
    font-size: 30px;
}

.btn-md {
    font-size: 34px; /* Adjust the font size to make the button bigger
*/
    padding: 10px 20px; /* Adjust padding for width and height */
}

.nav-item {
    list-style-type: none;
}

    header {
        background-color: #333;
        color: white;
        padding: 10px;
        text-align: center;
    }

    nav {
        display: flex;
        justify-content: space-around;
        background-color: #444;
        padding: 10px;
    }

    nav a {
        color: white;
        text-decoration: none;
    }

```

```

    }

    nav a:hover {
        text-decoration: underline;
    }

</style>
</head>
<header>
    <nav>
        <a href="TalentForgeHomePage.php">Home</a>
        <a href="LoginPage.php">Login</a>
        <a href="About.php">About</a>
    </nav>
</header>
<body>
    <h1>TalentForge</h1>
    

    <div id="description">
        <div class="tile">
            <p><strong>Unlock Your Potential with Personalized
CVs:</strong> TalentForge empowers users to effortlessly input their
personal and professional details, creating customized CVs or resumes using
a variety of default company templates. Craft professional profiles that
stand out and streamline your job application process.</p>
        </div>

        <div class="tile">
            <p><strong>Seamless CV Download and Sharing:</strong>
TalentForge enables users to download and save their CVs in digital
formats, facilitating easy sharing with potential employers. Say goodbye to
printed materials and physical distribution—disseminate job-related
information efficiently in the digital realm.</p>
        </div>

        <div class="tile">
            <p><strong>Talent Repository for Streamlined
Recruitment:</strong> TalentForge introduces the Talent Box, a repository
where users voluntarily share their CVs. Company users access and review
this centralized database, simplifying talent acquisition. Elevate your
recruitment process with a diverse pool of potential candidates.</p>
        </div>

        <div class="tile">
            <p><strong>Enhanced Company User Functionality:</strong>
Tailored for company users, TalentForge offers a dedicated login space.
Create and customize CV templates aligned with your organization's
requirements, fostering brand consistency and efficiency in the hiring
process.</p>
        </div>

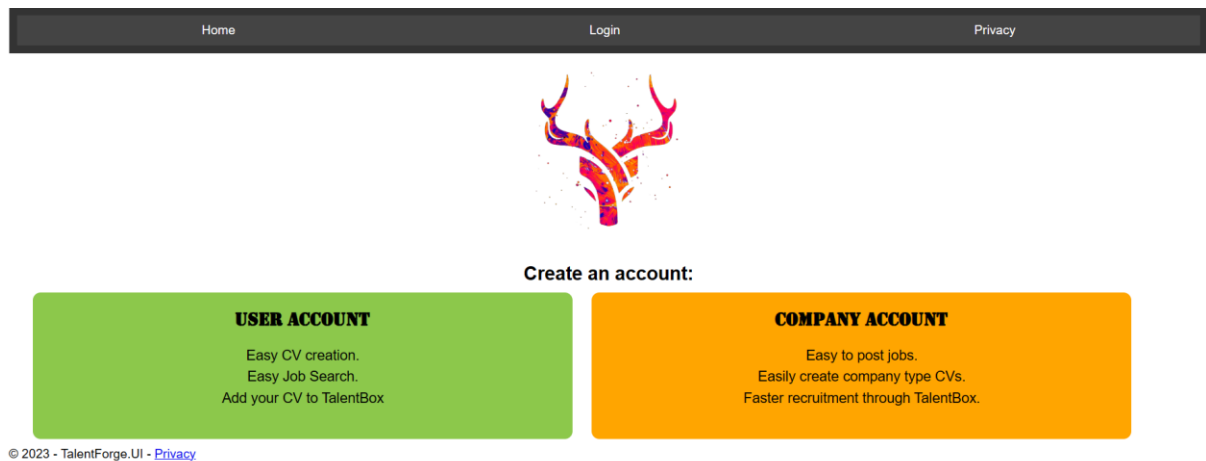
        <div class="tile">
            <p><strong>Job Advertisement and Seamless Interaction:</strong>
TalentForge's "Jobs Tab" empowers company users to post job advertisements.
Regular users access these listings, applying for positions seamlessly.
Foster direct communication through an integrated emailing service,
providing valuable feedback on job applications.</p>
        </div>

        <div class="tile">

```

```
<p><strong>Embrace Environmental Friendliness:</strong>
TalentForge is committed to environmental sustainability. By reducing paper
waste associated with traditional CVs and application processes, it
operates in a digital environment, minimizing the need for physical
documents and lowering the environmental impact of job seeking and
recruitment travel.</p>
</div>
<div>
<p>&nbsp;&nbsp;&nbsp;</p>
</div>
</div>
</body>
</html>
```

CreateAccountHomePage:



```
<html lang="en"><head>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title> - TalentForge.UI</title>
  <link rel="stylesheet"
href="/lib/bootstrap/dist/css/bootstrap.min.css">
  <link rel="stylesheet"
href="/css/site.css?v=AKvNjO3dCPPS0eSU1Ez8T2wI280i08yGycV9ndytL-c">
  <link rel="stylesheet"
href="/TalentForge.UI.styles.css?v=BOevtv_u0DTDcc45o9LezKTwcQIkWJVzWz5aAxaD
Fhw">
</head>
<style>
  /* Global styles */
  body {
    font-family: Arial, sans-serif;
  }

  /* Style for the buttons */
  .huge-button {
    display: inline-block;
    width: 45%; /* Adjust the width as needed */
    margin: 10px;
```

```

padding: 20px;
text-align: center;
border: none;
border-radius: 10px;
font-size: 18px;
cursor: pointer;
position: relative; /* Add relative positioning */
left: 20px; /* Move the button 20px to the right */
}

/* Style for the headings */
.button-heading {
    font-family: 'Stencil', sans-serif;
    font-size: 24px;
    font-weight: bold;
    margin-bottom: 10px;
    color: black; /* Set text color to black */
}

/* Style for the text below the headings */
.button-text {
    line-height: 1.5;
    color: black; /* Set text color to black */
    margin-left: 1px; /* Adjust the margin for indentation */
    list-style-type: none; /* Remove the dot/point */
    padding-left: 0; /* Remove padding for the list */
}

/* Style for the User button */
.user-button {
    background-color: #8CC84B; /* Light green color */
    color: black; /* Black text */
}

/* Style for the Company button */
.company-button {
    background-color: #FFA500; /* Medium orange color */
    color: black; /* Black text */
}

/* Style for the centered image */
.centered-image {
    display: block;
    margin: 0 auto; /* Center the image */
    max-width: 100%; /* Ensure the image doesn't exceed its
container */
}

/* Style for the heading */
.account-heading {
    text-align: center;
    font-size: 24px;
    font-weight: bold;
    margin-top: 20px; /* Add margin to separate from the image */
    color: black; /* Set text color to black */
}

header {
    background-color: #333;
    color: white;
    padding: 10px;

```

```

        text-align: center;
    }

    nav {
        display: flex;
        justify-content: space-around;
        background-color: #444;
        padding: 10px;
    }

    nav a {
        color: white;
        text-decoration: none;
    }

    nav a:hover {
        text-decoration: underline;
    }
</style>
<body>

<header>
    <nav>
        <a href="TalentForgeHomePage.php">Home</a>
        <a href="LoginPage.php">Login</a>
        <a href="#Privacy">Privacy</a>
    </nav>

</header>
<div class="container">
    <main role="main" class="pb-3">

<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title></title>

<div>
    <!-- Centered Image -->
    

    <div class="account-heading">Create an account:</div>

    <!-- First Button - User Account -->
    <button class="huge-button user-button"
onclick="redirectTo('TalentForge/CreateRegAccount.php')">
        <div class="button-heading">User Account</div>
        <ul class="button-text">
            <li>Easy CV creation.</li>
            <li>Easy Job Search.</li>
            <li>Add your CV to TalentBox</li>
        </ul>
    </button>

    <!-- Second Button - Company Account -->

```



```

        <button class="huge-button company-button"
onclick="redirectTo('TalentForge/CreateCompAccount.php')">
        <div class="button-heading">Company Account</div>
        <ul class="button-text">
            <li>Easy to post jobs.</li>
            <li>Easily create company type CVs.</li>
            <li>Faster recruitment through TalentBox.</li>
        </ul>
    </button>

    <script>
        function redirectTo(action) {
            window.location.href = '/' + action;
        }
    </script>
</div>

<!-- Visual Studio Browser Link -->
<script type="text/javascript" src="/_vs/browserLink" async="async"
id="__browserLink_initializationData" data-
requestid="4c0ca29d30cc4786b4314calle09c51d" data-
requestmappingfromserver="false" data-
connecturl="http://localhost:57002/6f204c6b5f104e8c94ae5f34065da800/browser
Link"></script>
<!-- End Browser Link -->

<script src="/_framework/aspnetcore-browser-refresh.js"></script>

    </main>
</div>

<footer b-89rpdq017y="" class="border-top footer text-muted">
    <div b-89rpdq017y="" class="container">
        © 2023 - TalentForge.UI - <a href="/Home/Privacy">Privacy</a>
    </div>
</footer>
<script src="/lib/jquery/dist/jquery.min.js"></script>
<script src="/lib/bootstrap/dist/js/bootstrap.bundle.min.js"></script>
<script
src="/js/site.js?v=4q1jwFhaPaZgr8WAUSrux6hAuh0XDg9kPS3xIVq36I0"></script>

</body></html>

```

CreateCompAccount.php:

Create an company account

Company Registration Form

Name:

--

Surname:

--

Nationality:

--

ID Number:

Company:

Email:

Password:

Submit

```
<!DOCTYPE html>
<html>
<head>
    <style>
        body {
            font-family: Arial, sans-serif;
        }

        form {
            width: 600px;
            margin: 0 auto;
        }

        label {
            display: block;
            margin-top: 10px;
        }

        input[type="text"] {
            width: 100%;
            padding: 12px 20px;
            margin: 8px 0;
```

```

        box-sizing: border-box;
    }

    button[type="submit"] {
        background-color: #4CAF50;
        color: white;
        padding: 14px 20px;
        margin: 8px 0;
        border: none;
        cursor: pointer;
        width: 100%;
    }

    button[type="submit"]:hover {
        opacity: 0.8;
    }
</style>
</head>
<body>

    <form method="post" asp-controller="Home" asp-action="Login"
action="ReturnToLoginComp.php">
        <h1>Company Registration Form</h1>

        <label for="name">Name:</label>
        <input type="text" id="Name" name="Name" required>

        <label for="surname">Surname:</label>
        <input type="text" id="Surname" name="Surname" required>

        <label for="nationality">Nationality:</label>
        <input type="text" id="Nationality" name="Nationality" required>

        <label for="idnumber ">ID Number:</label>
        <input type="text" id="IDNumber" name="IDNumber" required>

        <label for="company">Company:</label>
        <input type="text" id="Company" name="Company" required>

        <label for="company">Email:</label>
        <input type="text" id="Email" name="Email" required>

        <label for="company">Password:</label>
        <input type="text" id="Password" name="Password" required>

        <button type="submit">Submit</button>
    </form>

</body>
</html>

```

CreateRegAccount.php:

Create an account for regular user

Title 1: User Information

Name:

Surname:

Gender : ☐ Male ☐ Female

Marital Status: ☐ Single ☐ Married

Contact Number:

Email:

Password :

Residential Address:

Postal Address:

ID Number:

```
<?php
error_reporting(0);

?>
<!DOCTYPE html>
<html lang="en">
<head>

<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<style>
    body {
        font-family: Arial, sans-serif;
        background-color: #f4f4f4;
        margin: 0;
```

```

        padding: 0;
    }

    .container {
        width: 80%;
        margin: auto;
        overflow: hidden;
    }

    .tile {
        background-color: #fff;
        box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
        margin: 20px;
        padding: 20px;
        border-radius: 5px;
    }

    label {
        display: block;
    }

    input,
    select {
        width: 50%;
        padding: 10px;
        margin-bottom: 15px;
        display: inline-block;
        border: 1px solid #ccc;
        box-sizing: border-box;
        border-radius: 10px;
    }

    input[type="submit"] {
        background-color: #4caf50;
        color: white;
        padding: 10px 15px;
        border: none;
        border-radius: 4px;
        cursor: pointer;
    }

    input[type="submit"]:hover {
        background-color: #45a049;
    }

</style>
<title>Form</title>
</head>
<body>
    <div class="container">
        <div class="tile">
            <h2>Tile 1: User Information</h2>
            <form name="Personaldetails" action="skills.php" method="POST">
                <label for="name">Name:</label>
                <input type="text" id="name" name="name" required>

                <label for="surname">Surname:</label>

```

```

        <input type="text" id="surname" name="surname" required>

        <label for="gender"> Gender : </label>
        <Input type = 'Radio' Name ='gender' value='male'
>Male<Input type = 'Radio' Name ='gender' value= 'female' required>Female

        <label for="MaritalStatus">Marital Status: </label>
        <Input type = 'Radio' Name ='MaritalStatus'
value='Single'>Single<Input type = 'Radio' Name ='MaritalStatus' value=
'Married' required>Married

        <label for="contactNumber">Contact Number:</label>
        <input type="text" id="contactNumber" name="contactNumber"
required>

        <label for="email">Email:</label>
        <input type="text" id="email" name="email" required>

        <label for="password">Password :</label>
        <input type="text" id="pass" name="pass" required>

        <label for="residentialAddress">Residential
Address:</label>
        <input type="text" id="residentialAddress"
name="residentialAddress" required>

        <label for="postalAddress">Postal Address:</label>
        <input type="text" id="postalAddress" name="postalAddress"
required>

        <label for="IDNumber">ID Number:</label>
        <input type="text" id="IDNumber" name="IDNumber" required>

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

    </form>
</div>
</div>
</body>
</html>
<?php
?>

```

The data is sent to be inserted in skills.php

Skills.php:

Tile 2: Skills

Skill 1:

Skill 2:

Submit

```
<?php
// Connection to database
$servername="localhost";
$username="root";
$password ="";
$db="talentforge";
$conn = mysqli_connect($servername, $username, $password, $db);
if (!$conn){
die("Connection failed:" . mysqli_connect_error ());
}

function containsOnlyLetters($str) {
    return preg_match('/^[a-zA-Z]+$/', $str);
}

// Function to check if a string contains only numbers
function containsOnlyNumbers($str) {
    return is_numeric($str);
}

// Function to check if a string has a specific length
function hasSpecificLength($str, $length) {
    return strlen($str) == $length;
}

function isValidEmail($email) {
    return filter_var($email, FILTER_VALIDATE_EMAIL);
}

function isValidPassword($password) {
    // Check if the password is at least 8 characters long
    if (strlen($password) < 8) {
        return false;
    }

    // Check if the password contains at least one uppercase letter
    if (!preg_match('/[A-Z]/', $password)) {
        return false;
    }

    // Check if the password contains at least one lowercase letter
    if (!preg_match('/[a-z]/', $password)) {
        return false;
    }

    // Check if the password contains at least one digit
    if (!preg_match('/[0-9]/', $password)) {
        return false;
    }
}
```

```

        // Check if the password contains at least one special character
        if (!preg_match('/[!@#$%^&*(),.?":{}|<>]/', $password)) {
            return false;
        }

        // If all conditions pass, the password is valid
        return true;
    }

    // Check if the form is submitted
    if ($_SERVER["REQUEST_METHOD"] == "POST") {
        // Retrieve username and password from the form
        $name = $_POST["name"];
        $surname = $_POST["surname"];
        $gender = $_POST["gender"];
        $maritalStatus = $_POST["MaritalStatus"];
        $contactNumber = $_POST["contactNumber"];
        $email = $_POST["email"];
        $pass = $_POST["pass"];
        $residentialAddress = $_POST["residentialAddress"];
        $postalAddress = $_POST["postalAddress"];
        $IDNumber = $_POST["IDNumber"];

        // Validate and sanitize user inputs (you should enhance this based on
        your requirements)
        $name = mysqli_real_escape_string($conn, $name);
        $surname = mysqli_real_escape_string($conn, $surname);
        $gender = mysqli_real_escape_string($conn, $gender);
        $maritalStatus = mysqli_real_escape_string($conn, $maritalStatus);
        $contactNumber = mysqli_real_escape_string($conn, $contactNumber);
        $email = mysqli_real_escape_string($conn, $email);
        $username = mysqli_real_escape_string($conn, $username);
        $pass = mysqli_real_escape_string($conn, $pass);
        $residentialAddress = mysqli_real_escape_string($conn,
        $residentialAddress);
        $postalAddress = mysqli_real_escape_string($conn, $postalAddress);
        $IDNumber = mysqli_real_escape_string($conn, $IDNumber);

        $sql = "INSERT INTO tblregusers
        (Name,Surname,Gender,MaritalStatus,ContactNum,Email>Password,ResAddress,Pos
        talAddress,IDNumber) VALUES
        ('$name', '$surname', '$gender', '$maritalStatus', '$contactNumber',
        '$email', '$pass', '$residentialAddress', '$postalAddress', '$IDNumber')";

        if (mysqli_query($conn, $sql)) {
            echo '<script>alert("Registration successfully");</script>';

            } else {
                echo '<script>alert("Unsuccessfully Registration Try
                again");</script>';
            }

        }

    }

    //RUID
    if ($_SERVER["REQUEST_METHOD"] == "POST") {
        // Retrieve username and password from the form

```



```

// Query to search for the user
$sql = "SELECT RUID FROM tblregusers WHERE Email = '$email' AND Password =
'$pass'";
$result = $conn->query($sql);

// Check if a matching record was found
if ($result->num_rows > 0) {
    // Fetch the id from the result
    $row = $result->fetch_assoc();
    $ruid = $row['RUID'];
    echo "Login successful! RUID: $ruid";
}
}

// Close connection
$conn->close();// Close connection
?>

```

```

<!DOCTYPE html>
<html lang="en">
<head>

<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<style>
    body {
        font-family: Arial, sans-serif;
        background-color: #f4f4f4;
        margin: 0;
        padding: 0;
    }

    .container {
        width: 80%;
        margin: auto;
        overflow: hidden;
    }

    .tile {
        background-color: #fff;
        box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
        margin: 20px;
        padding: 20px;
        border-radius: 5px;
    }

    label {
        display: block;
    }

    input,
    select {
        width: 50%;
        padding: 10px;
        margin-bottom: 15px;
        display: inline-block;
        border: 1px solid #ccc;
    }

```

```

        box-sizing: border-box;
        border-radius: 10px;
    }

    input[type="submit"] {
        background-color: #4caf50;
        color: white;
        padding: 10px 15px;
        border: none;
        border-radius: 4px;
        cursor: pointer;
    }

    input[type="submit"]:hover {
        background-color: #45a049;
    }

</style>
<title>Form</title>
</head>
<body>
        <form name="skillsdetails"
action="SchoolInformation.php" method="POST" enctype="multipart/formdata">

        <div class="tile">
            <h2>Tile 2: Skills</h2>
            <label for="skill1">Skill 1:</label>
            <input type="text" id="skill1" name="skill1">
            <label for="skill2">Skill 2:</label>
            <input type="text" id="skill2" name="skill2">
            <input type="hidden" name="email" value="<?php echo
$email;?>">
            <input type="hidden" name="password" value="<?php echo
$password;?>">
            <input type="hidden" name="ruid" value="<?php echo
$ruid;?>">
            <input type="submit" value="Submit">
        </form>
    </div>
</div>
</body>
</html>

<?php
error_reporting(0);

?>

```

Skills data is sent and captured in SchoolInformation.php

SchoolInformation.php:

```
<?php
// Connection to database
$servername="localhost";
$username="root";
$password ="";
$db="talentforge";
$conn = mysqli_connect($servername, $username, $password, $db);
if (!$conn){
die("Connection failed:" . mysqli_connect_error ());
}

// Check if the form is submitted
if ($_SERVER["REQUEST_METHOD"] == "POST") {
    // Retrieve username and password from the form
    $email = $_POST["email"];
    $pass = $_POST["password"];
    $skill1 = $_POST["skill1"];
    $skill2 = $_POST["skill2"];
    $ruid = $_POST["ruid"];
    // Validate and sanitize user inputs (you should enhance this based on
    your requirements)
    $skill1 = mysqli_real_escape_string($conn, $skill1);
    $skill2 = mysqli_real_escape_string($conn, $skill2);

    $sql = "INSERT INTO tblskills (RUID,Skill1,Skill2) VALUES
    ('$ruid', '$skill1', '$skill2')";

    if (mysqli_query($conn, $sql)) {
        echo '<script>alert("Registration successfully");</script>';
    } else {
        echo '<script>alert("Unsuccessfully Registration Try
again");</script>';
    }
}
$conn->close();// Close connection
?>
```

```
<!DOCTYPE html>
<html lang="en">
<head>

<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<style>
    body {
        font-family: Arial, sans-serif;
        background-color: #f4f4f4;
        margin: 0;
        padding: 0;
    }

    .container {
        width: 80%;
        margin: auto;
        overflow: hidden;
    }
```

```

.tile {
    background-color: #fff;
    box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
    margin: 20px;
    padding: 20px;
    border-radius: 5px;
}

label {
    display: block;
}

input,
select {
    width: 50%;
    padding: 10px;
    margin-bottom: 15px;
    display: inline-block;
    border: 1px solid #ccc;
    box-sizing: border-box;
    border-radius: 10px;
}

input[type="submit"] {
    background-color: #4caf50;
    color: white;
    padding: 10px 15px;
    border: none;
    border-radius: 4px;
    cursor: pointer;
}

input[type="submit"]:hover {
    background-color: #45a049;
}

</style>
<title>Form</title>
</head>
<body>
    <form name="SchoolInformationdetails"
action="HigherEducation.php" method="POST">

    <div class="tile">
    <h2>Tile 4: School Information</h2>
    <label for="lastSchool">Last School Attended:</label>
    <input type="text" id="lastSchool" name="lastSchool">

    <label for="highestGrade">Highest Grade Passed:</label>
    <input type="text" id="highestGrade" name="highestGrade">

    <label for="finalYear">Final Year:</label>
    <input type="text" id="finalYear" name="finalYear">

    <label for="subject1">Subject 1:</label>
    <input type="text" id="subject1" name="subject1" required>

```

```

<label for="mark1">Mark 1:</label>
<input type="text" id="mark1" name="mark1" required>

<label for="subject2">Subject 2:</label>
<input type="text" id="subject2" name="subject2" required>

<label for="mark2">Mark 2:</label>
<input type="text" id="mark2" name="mark2" required>

<label for="subject3">Subject 3:</label>
<input type="text" id="subject3" name="subject3" required>

<label for="mark3">Mark 3:</label>
<input type="text" id="mark3" name="mark3" required>

<label for="subject1">Subject 4:</label>
<input type="text" id="subject4" name="subject4" required>

<label for="mark4">Mark 4:</label>
<input type="text" id="mark4" name="mark4" required>

<label for="subject2">Subject 5:</label>
<input type="text" id="subject5" name="subject5" required>

<label for="mark2">Mark 5:</label>
<input type="text" id="mark5" name="mark5" required>

<!-- Subject 3 -->
<label for="subject3">Subject 6:</label>
<input type="text" id="subject6" name="subject6" required>

<label for="mark3">Mark 6:</label>
<input type="text" id="mark6" name="mark6" required>

<!-- Subject 1 -->
<label for="subject1">Subject 7:</label>
<input type="text" id="subject7" name="subject7" required>

<label for="mark1">Mark 7:</label>
<input type="text" id="mark7" name="mark7" required>
    <input type="hidden" name="email" value="<?php echo
$email;?>">
    <input type="hidden" name="password" value="<?php echo
$password;?>">
    <input type="hidden" name="ruid" value="<?php echo $ruid;?>">
</div>

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

</form>
</div>
</div>
</body>
</html>

<?php
error_reporting(0);
?>

```

School information is sent and captured in Higered.php

HigherEducation.php

College Name and Location:

Field Of Study:

Active:

☐ Yes

☐ NO

Completed:

☐ Yes

☐ NO

Submit

```
<?php
// Connection to database
$servername="localhost";
$username="root";
$password ="";
$db="talentforge";
$conn = mysqli_connect($servername, $username, $password, $db);
if (!$conn){
die("Connection failed:" . mysqli_connect_error ());
}

// Check if the form is submitted
if ($_SERVER["REQUEST_METHOD"] == "POST") {
    // Retrieve username and password from the form
    $email = $_POST["email"];
    $pass = $_POST["password"];
    $ruid = $_POST["ruid"];
    $lastSchool = $_POST["lastSchool"];
    $highestGrade = $_POST["highestGrade"];
    $finalYear = $_POST["finalYear"];
    $sub1 = $_POST["subject1"];
    $mrk1 = $_POST["mark1"];
    $sub2 = $_POST["subject2"];
    $mrk2 = $_POST["mark2"];
    $sub3 = $_POST["subject3"];
    $mrk3 = $_POST["mark3"];
    $sub4 = $_POST["subject4"];
    $mrk4 = $_POST["mark4"];
    $sub5 = $_POST["subject5"];
    $mrk5 = $_POST["mark5"];
    $sub6 = $_POST["subject6"];
    $mrk6 = $_POST["mark6"];
    $sub7 = $_POST["subject7"];
    $mrk7 = $_POST["mark7"];
    // Validate and sanitize user inputs (you should enhance this based on
    your requirements)
    $lastSchool = mysqli_real_escape_string($conn, $lastSchool);
```

```

$highestGrade = mysqli_real_escape_string($conn, $highestGrade);
$finalYear = mysqli_real_escape_string($conn, $finalYear);
$sub1 = mysqli_real_escape_string($conn, $sub1);
$mrk1 = mysqli_real_escape_string($conn, $mrk1);
$sub2 = mysqli_real_escape_string($conn, $sub2);
$mrk2 = mysqli_real_escape_string($conn, $mrk2);
$sub3 = mysqli_real_escape_string($conn, $sub3);
$mrk3 = mysqli_real_escape_string($conn, $mrk3);
$sub4 = mysqli_real_escape_string($conn, $sub4);
$mrk4 = mysqli_real_escape_string($conn, $mrk4);
$sub5 = mysqli_real_escape_string($conn, $sub5);
$mrk5 = mysqli_real_escape_string($conn, $mrk5);
$sub6 = mysqli_real_escape_string($conn, $sub6);
$mrk6 = mysqli_real_escape_string($conn, $mrk6);
$sub7 = mysqli_real_escape_string($conn, $sub7);
$mrk7 = mysqli_real_escape_string($conn, $mrk7);

$sql = "INSERT INTO tblschoolinfo
(RUID,LastSchool,highestGrade,finalYear,
Subject1, Mark1, Subject2, Mark2, Subject3, Mark3, Subject4, Mark4,
Subject5, Mark5,
Subject6, Mark6, Subject7, Mark7)
VALUES ('$ruid', '$lastSchool', '$highestGrade', '$finalYear',
'$sub1', '$mrk1', '$sub2', '$mrk2',
'$sub3', '$mrk3', '$sub4', '$mrk4', '$sub5', '$mrk5', '$sub6',
'$mrk6', '$sub7', '$mrk7')";

if (mysqli_query($conn, $sql)) {
    echo '<script>alert("Registration
successfully");</script>';
} else {
    echo '<script>alert("Unsuccessfully Registration Try
again");</script>';
}
}
$conn->close();// Close connection
?>
<!DOCTYPE html>
<html lang="en">
<head>

<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<style>
    body {
        font-family: Arial, sans-serif;
        background-color: #f4f4f4;
        margin: 0;
        padding: 0;
    }

    .container {
        width: 80%;
        margin: auto;
        overflow: hidden;
    }

    .tile {
        background-color: #fff;

```

```

        box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
        margin: 20px;
        padding: 20px;
        border-radius: 5px;
    }

    label {
        display: block;
    }

    input,
    select {
        width: 50%;
        padding: 10px;
        margin-bottom: 15px;
        display: inline-block;
        border: 1px solid #ccc;
        box-sizing: border-box;
        border-radius: 10px;
    }

    input[type="submit"] {
        background-color: #4caf50;
        color: white;
        padding: 10px 15px;
        border: none;
        border-radius: 4px;
        cursor: pointer;
    }

    input[type="submit"]:hover {
        background-color: #45a049;
    }

</style>
<title>Form</title>
</head>
<body>

    <form name="HigerEducation" action="EmployeeHistory.php" method="POST"
    >

        <label for="collegeName">College Name and
Location:</label>
        <input type="text" id="collegeName" name="collegeName">

        <label for="fieldOfStudy">Field Of Study:</label>
        <input type="text" id="fieldOfStudy"
name="fieldOfStudy">

        <label for="Active">Active: </label>
        <input type = 'Radio' Name ='Active' value='Yes'>Yes<input
type = 'Radio' Name ='Active' value= 'NO'>NO

```



```

        <label for="Completed">Completed: </label>
        <Input type = 'Radio' Name ='Completed'
value='Yes'>Yes<Input type = 'Radio' Name ='Completed' value= 'NO'>NO

        <input type="hidden" name="email" value="<?php echo
$email;?>">
        <input type="hidden" name="password" value="<?php
echo $password;?>">
        <input type="hidden" name="ruid" value="<?php echo
$ruid;?>">

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

    </form>

</body>
</html>

```

HigherEducation is sent and captured in EmployeeHistory.php

EmployeeHistory.php:

Company Name:

Position:

Reference:

Reference Contact:

```

<?php
// Connection to database
$servername="localhost";
$username="root";
$password ="";
$db="talentforge";
$conn = mysqli_connect($servername, $username, $password, $db);
if (!$conn){
die("Connection failed:" . mysqli_connect_error ());
}

// Check if the form is submitted
if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $email = $_POST["email"];
    $pass = $_POST["password"];
    $ruid = $_POST["ruid"];
    // Retrieve username and password from the form
    $collegeName = $_POST["collegeName"];
}

```

```

$fieldOfStudy = $_POST["fieldOfStudy"];
$active = $_POST["Active"];
$completed = $_POST["Completed"];
// Validate and sanitize user inputs (you should enhance this based on
your requirements)
$collegeName = mysqli_real_escape_string($conn, $collegeName);
$fieldOfStudy = mysqli_real_escape_string($conn, $fieldOfStudy);
$active = mysqli_real_escape_string($conn, $active);
$completed = mysqli_real_escape_string($conn, $completed);

$sql = "INSERT INTO tblhighered
(RUID,collegeName,fieldOfStudy,Active,Completed)
VALUES ('$ruid','$collegeName', '$fieldOfStudy', '$active',
'$completed');"

if (mysqli_query($conn, $sql)) {
    echo '<script>alert("Registration
successfully");</script>';

} else {
    echo '<script>alert("Unsuccessfully Registration Try
again");</script>';

}
$conn->close();// Close connection
?>
<!DOCTYPE html>
<html lang="en">
<head>

<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<style>
    body {
        font-family: Arial, sans-serif;
        background-color: #f4f4f4;
        margin: 0;
        padding: 0;
    }

    .container {
        width: 80%;
        margin: auto;
        overflow: hidden;
    }

    .tile {
        background-color: #fff;
        box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
        margin: 20px;
        padding: 20px;
        border-radius: 5px;
    }

    label {
        display: block;
    }

    input,

```

```

        select {
            width: 50%;
            padding: 10px;
            margin-bottom: 15px;
            display: inline-block;
            border: 1px solid #ccc;
            box-sizing: border-box;
            border-radius: 10px;
        }

        input[type="submit"] {
            background-color: #4caf50;
            color: white;
            padding: 10px 15px;
            border: none;
            border-radius: 4px;
            cursor: pointer;
        }

        input[type="submit"]:hover {
            background-color: #45a049;
        }

    </style>
    <title>Form</title>
</head>
<body>
        <form name="PersonalDetails" action="ReturnToLogin.php"
method="POST">
            <label for="companyName">Company Name:</label>
            <input type="text" id="companyName" name="companyName">

            <label for="position">Position:</label>
            <input type="text" id="position" name="position">

            <label for="reference">Reference:</label>
            <input type="text" id="reference" name="reference">

            <label for="referenceContact">Reference
Contact:</label>
            <input type="text" id="referenceContact"
name="referenceContact">

            <input type="hidden" name="ruid" value="<?php echo
$ruid;?>">
            <input type="submit" value="Submit">

        </div>
    </form>
</div>
</body>
</html>

<?php
// Connection to database
$servername="localhost";

```

```

$username="root";
$password ="";
$db="talentforge";
$conn = mysqli_connect($servername, $username, $password, $db);
if (!$conn){
die("Connection failed:" . mysqli_connect_error ());
}

// Check if the form is submitted
if ($_SERVER["REQUEST_METHOD"] == "POST") {

    $ruid = $_POST["ruid"];
    // Retrieve username and password from the form
    $companyName = $_POST["companyName"];
    $position = $_POST["position"];
    $reference = $_POST["reference"];
    $referenceContact = $_POST["referenceContact"];
    // Validate and sanitize user inputs (you should enhance this based on
    your requirements)
    $companyName = mysqli_real_escape_string($conn, $companyName);
    $position = mysqli_real_escape_string($conn, $position);
    $reference = mysqli_real_escape_string($conn, $reference);
    $referenceContact = mysqli_real_escape_string($conn,
    $referenceContact);

    $sql ="INSERT INTO tblemployeehistory (RUID,Company, Position,
    Reference, ReferenceContact)
    VALUES ('$ruid','$companyName', '$position', '$reference',
    '$referenceContact');"

    if (mysqli_query($conn, $sql)) {
        echo '<script>alert("successfully");</script>';

    } else {
        echo '<script>alert("Unsuccessfully Registration
    Try again");</script>';
    }
}
$conn->close();// Close connection
?>

```

LoginPage.php

```

<html lang="en"><head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title> - TalentForge.UI</title>
    <link rel="stylesheet"
href="/lib/bootstrap/dist/css/bootstrap.min.css">
    <link rel="stylesheet"
href="/css/site.css?v=AKvNjO3dCPPS0eSU1Ez8T2wI280i08yGycV9ndytL-c">

```

```

    <link rel="stylesheet"
href="/TalentForge.UI.styles.css?v=BOevtv_u0DTDcc45o9LezKTwcQIkWJVzWz5aAxaD
Fhw">
</head>
<style>
    /* Global styles */
    body {
        font-family: Arial, sans-serif;
    }

    /* Style for the buttons */
    .huge-button {
        display: inline-block;
        width: 45%; /* Adjust the width as needed */
        margin: 10px;
        padding: 20px;
        text-align: center;
        border: none;
        border-radius: 10px;
        font-size: 18px;
        cursor: pointer;
        position: relative; /* Add relative positioning */
        left: 20px; /* Move the button 20px to the right */
    }

    /* Style for the headings */
    .button-heading {
        font-family: 'Stencil', sans-serif;
        font-size: 24px;
        font-weight: bold;
        margin-bottom: 10px;
        color: black; /* Set text color to black */
    }

    /* Style for the text below the headings */
    .button-text {
        line-height: 1.5;
        color: black; /* Set text color to black */
        margin-left: 1px; /* Adjust the margin for indentation */
        list-style-type: none; /* Remove the dot/point */
        padding-left: 0; /* Remove padding for the list */
    }

    /* Style for the User button */
    .user-button {
        background-color: #8CC84B; /* Light green color */
        color: black; /* Black text */
    }

    /* Style for the Company button */
    .company-button {
        background-color: #FFA500; /* Medium orange color */
        color: black; /* Black text */
    }

    /* Style for the centered image */
    .centered-image {
        display: block;
        margin: 0 auto; /* Center the image */
        max-width: 100%; /* Ensure the image doesn't exceed its
container */

```

```

    }

    /* Style for the heading */
    .account-heading {
        text-align: center;
        font-size: 24px;
        font-weight: bold;
        margin-top: 20px; /* Add margin to separate from the image */
        color: black; /* Set text color to black */
    }

    header {
        background-color: #333;
        color: white;
        padding: 10px;
        text-align: center;
    }

    nav {
        display: flex;
        justify-content: space-around;
        background-color: #444;
        padding: 10px;
    }

    nav a {
        color: white;
        text-decoration: none;
    }

    nav a:hover {
        text-decoration: underline;
    }
</style>
<body>

<header>
    <nav>
        <a href="TalentForgeHomePage.php">Home</a>
        <a href="LoginPage.php">Login</a>
        <a href="About.php">About</a>
    </nav>

</header>
<div class="container">
    <main role="main" class="pb-3">

<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title></title>

<div>
    <!-- Centered Image -->
    

```

```

        <div class="account-heading">Choose a login type:</div>

        <!-- First Button - User Account -->
        <button class="huge-button user-button"
onclick="redirectTo('TalentForge/LoginRegUser.php') ">
            <div class="button-heading">User Account</div>

        </button>

        <!-- Second Button - Company Account -->
        <button class="huge-button company-button"
onclick="redirectTo('TalentForge/LoginCompUser.php') ">
            <div class="button-heading">Company Account</div>

        </button>
        Redirect to user login or company login
        <script>
            function redirectTo(action) {
                window.location.href = '/' + action;
            }
        </script>
    </div>

    <!-- Visual Studio Browser Link -->
    <script type="text/javascript" src="/_vs/browserLink" async="async"
id="__browserLink_initializationData" data-
requestid="4c0ca29d30cc4786b4314calle09c51d" data-
requestmappingfromserver="false" data-
connecturl="http://localhost:57002/6f204c6b5f104e8c94ae5f34065da800/browser
Link"></script>
    <!-- End Browser Link -->

    <script src="/_framework/aspnetcore-browser-refresh.js"></script>


    </main>
</div>

<footer b-89rpdq017y="" class="border-top footer text-muted">
    <div b-89rpdq017y="" class="container">
        © 2023 - TalentForge.UI - <a href="/Home/Privacy">Privacy</a>
    </div>
</footer>
<script src="/lib/jquery/dist/jquery.min.js"></script>
<script src="/lib/bootstrap/dist/js/bootstrap.bundle.min.js"></script>
<script
src="/js/site.js?v=4q1jwFhaPaZgr8WAUSrux6hAuh0XDg9kPS3xIVq36I0"></script>

</body></html>

```

[Home](#)[Login](#)[About](#)



Choose a login type:

USER ACCOUNT

COMPANY ACCOUNT

© 2023 - TalentForge.UI - [Privacy](#)

Userlogin.php

User Login



Email:

Password:

Login

[Create an account](#)

This code verifies the user login details in the database and stores their RUID in a sessions for future use if the login is incorrect user is redirected to the loginpage else is redirected to RUHome:

```
<?php

session_start(); // Start the session

//Connection to database

$servername="localhost";

$username="root";

$password ="";

$db="talentforge";

$conn = mysqli_connect($servername, $username, $password, $db);

if (!$conn){

die("Connection fa1led:" . mysqli_connect_error ());

}


if ($_SERVER["REQUEST_METHOD"] == "POST") {

    // Retrieve username and password from the form

    $username = $_POST["Username"];

    $password = $_POST["Password"];


    // Perform validation and authentication (replace this with your actual authentication logic)

    if (isValidCredentials($username, $password)) {

        // Retrieve RUID from the database

        $ruidResult = $conn->query("SELECT RUID FROM tblregusers WHERE Email = '$username'");

        $ruidRow = $ruidResult->fetch_assoc();

        $ruid = $ruidRow['RUID'];


        // Store RUID in session variable For future

        $_SESSION['ruid'] = $ruid;


        // Redirect to User.php
```

```

        echo '<script>alert("Successfully");</script>';

        header("Location: RUHome.php");

        exit();
    } else {

        // Show error message and redirect to login.php

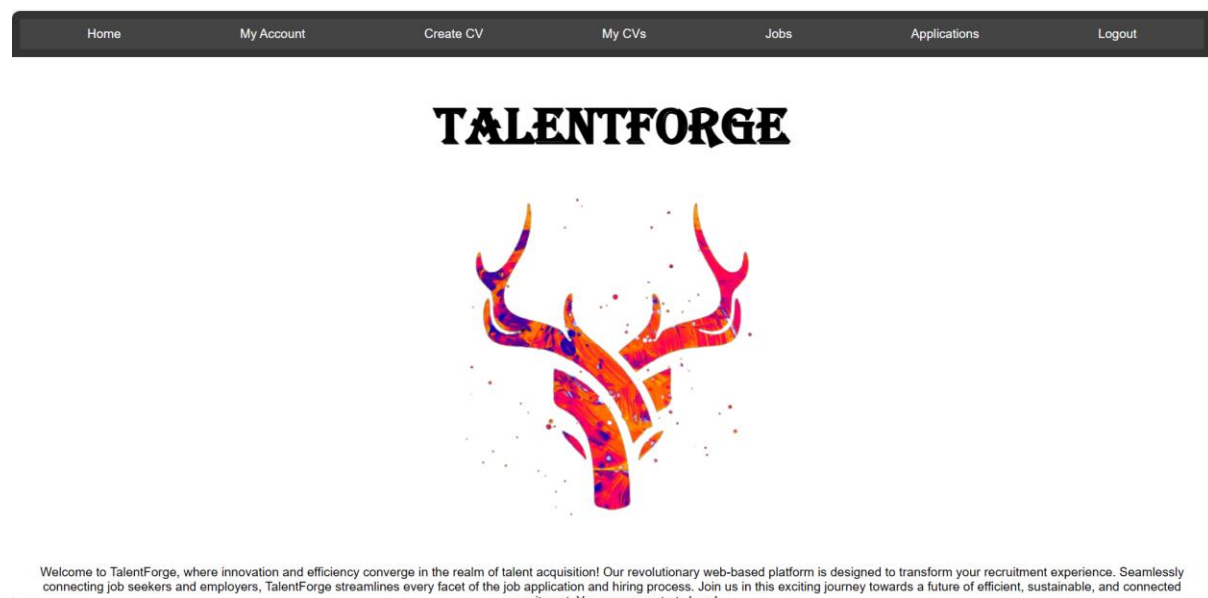
        echo '<script>alert("Unsuccessfully");</script>';

        header("Location: LoginPage.php");

        exit();
    }
}

```

RUHome.php



My Account page:

This show all the users information and allows them to update their information as well as delete their account

Home	My Account	Create CV	My CVs	Jobs	Applications	Logout
------	------------	-----------	--------	------	--------------	--------

My Account

Name:

Surname:

ID Number:

Nationality:

Marital Status:

Contact Number:

Email:

```
<?php
session_start(); // Start the session

// Check if RUID is set in the session
if (isset($_SESSION['ruid'])) {
    $ruid = $_SESSION['ruid'];

    // Connection to the database
    $servername = "localhost";
    $username = "root";
    $password = "";
    $db = "talentforge";
    $conn = mysqli_connect($servername, $username, $password, $db);
    if (!$conn) {
        die("Connection failed: " . mysqli_connect_error());
    }

    // Fetch user data from tblcompusers based on RUID
    $result = $conn->query("SELECT * FROM tblregusers WHERE RUID = '$ruid'");
    if ($result->num_rows > 0) {
        $userData = $result->fetch_assoc();
    } else {
        // Handle error or redirect to login page
        header("Location: LoginPage.php");
        exit();
    }
}
```

Code to display all current information in to the form

```
// Fetch additional data from other tables
$skillsResult = $conn->query("SELECT * FROM tblskills WHERE RUID = '$ruid'");
$skillsData = $skillsResult->fetch_assoc();

$schoolInfoResult = $conn->query("SELECT * FROM tblschoolinfo WHERE RUID = '$ruid'");
$schoolInfoData = $schoolInfoResult->fetch_assoc();
```

```

        $higheredResult = $conn->query("SELECT * FROM tblhighered WHERE RUID = '$ruid'");
        $higheredData = $higheredResult->fetch_assoc();

        $employeeHistoryResult = $conn->query("SELECT * FROM tblemployeeehistory WHERE RUID = '$ruid'");
        $employeeHistoryData = $employeeHistoryResult->fetch_assoc();

        // ... (Repeat for other tables)

```

CODE to delete the user account

```

// Check if the form is submitted
if ($_SERVER["REQUEST_METHOD"] == "POST") {
    if (isset($_POST['deleteAccount']) && $_POST['deleteAccount'] == 1)
    {
        // Delete account logic here
        $deleteSkillsQuery = "DELETE FROM tblskills WHERE RUID = '$ruid'";
        $deleteSchoolQuery = "DELETE FROM tblschoolinfo WHERE RUID = '$ruid'";
        $deleteHigherEdQuery = "DELETE FROM tblhighered WHERE RUID = '$ruid'";
        $deleteEmpHistQuery = "DELETE FROM tblemployeeehistory WHERE RUID = '$ruid'";
        $deleteUserQuery = "DELETE FROM tblregusers WHERE RUID = '$ruid'";

        try {
            // Execute the delete queries
            $conn->query($deleteUserQuery);
            $conn->query($deleteSkillsQuery);
            $conn->query($deleteSchoolQuery);
            $conn->query($deleteHigherEdQuery);
            $conn->query($deleteEmpHistQuery);

            // Successful delete
            echo "Account deleted successfully.";
            header("Location: LoginPage.php");
            exit();
        } catch (Exception $e) {
            // Handle error
            echo "Error deleting account: " . $e->getMessage();
            header("Location: RegUserAccountPage.php");
            exit();
        }
    } else {
        // Update logic here (replace the placeholder with your actual update logic)
        $conn->begin_transaction();

```

Code to update all information

```

        // Fetch updated values from POST data
        $updatedName = $_POST['Name'];
        $updatedSurname = $_POST['Surname'];
        $updatedIDNumber = $_POST['IDNumber'];
        $updatedGender = $_POST['Gender'];
        $updatedMaritalStatus = $_POST['MaritalStatus'];
        $updatedContactNum = $_POST['ContactNum'];

```

```

$updatedEmail = $_POST['Email'];
$updatedResAddress = $_POST['ResAddress'];
$updatedPostalAddress = $_POST['PostalAddress'];
$updatedPassword = $_POST['Password'];
$updatedSkill1 = $_POST['skill1'];
$updatedSkill2 = $_POST['skill2'];
$updatedLastSchool = $_POST['LastSchool'];
$updatedHighestGrade = $_POST['highestGrade'];
$updatedFinalYear = $_POST['finalYear'];
$updatedSubject1 = $_POST['Subject1'];
$updatedMark1 = $_POST['Mark1'];
$updatedSubject2 = $_POST['Subject2'];
$updatedMark2 = $_POST['Mark2'];
$updatedSubject3 = $_POST['Subject3'];
$updatedMark3 = $_POST['Mark3'];
$updatedSubject4 = $_POST['Subject4'];
$updatedMark4 = $_POST['Mark4'];
$updatedSubject5 = $_POST['Subject5'];
$updatedMark5 = $_POST['Mark5'];
$updatedSubject6 = $_POST['Subject6'];
$updatedMark6 = $_POST['Mark6'];
$updatedSubject7 = $_POST['Subject7'];
$updatedMark7 = $_POST['Mark7'];
$updatedCollegeName = $_POST['collegeName'];
$updatedFieldOfStudy = $_POST['fieldOfStudy'];
$updatedActive = $_POST['Active'];
$updatedCompleted = $_POST['Completed'];
$updatedCompany = $_POST['Company'];
$updatedPosition = $_POST['Position'];
$updatedReference = $_POST['Reference'];
$updatedReferenceContact = $_POST['ReferenceContact'];

```

```

try {
    // Update the user data in the tblregusers table
    $updateUserQuery = "UPDATE tblregusers SET
        Name = '$updatedName',
        Surname = '$updatedSurname',
        IDNumber = '$updatedIDNumber',
        Gender = '$updatedGender',
        MaritalStatus = '$updatedMaritalstatus',
        ContactNum = '$updatedContactNum',
        Email = '$updatedEmail',
        ResAddress = '$updatedResAddress',
        PostalAddress = '$updatedPostalAddress'
        WHERE RUID = '$ruid'";

```

```

    // Update the skills table
    $updateSkillsQuery = "UPDATE skills SET
        skill1 = '$updatedSkill1',
        skill2 = '$updatedSkill2'
        WHERE RUID = '$ruid'";

```

```

    // Update the tblschoolinfo table
    $updateSchoolInfoQuery = "UPDATE tblschoolinfo SET
        LastSchool = '$updatedLastSchool',
        highestGrade = '$updatedHighestGrade',
        finalYear = '$updatedFinalYear'
        Subject1 = '$updatedSubject1',
        Mark1 = '$updatedMark1',
        Subject2 = '$updatedSubject2',
        Mark2 = '$updatedMark2',

```

```

Subject3='$updatedSubject3',
Mark3='$updatedMark3',
Subject4='$updatedSubject4',
Mark4='$updatedMark4',
Subject5='$updatedSubject5',
Mark5='$updatedMark5',
Subject6='$updatedSubject6',
Mark6='$updatedMark6',
Subject7='$updatedSubject7',
Mark7='$updatedMark7'
WHERE RUID = '$ruid';

```

```

$updateHigherEdQuery = "UPDATE tblhigherEd SET
collegeName='$updatedCollegeName',
fieldOfStudy='$updatedFieldOfStudy',
Active='$updatedActive',
Completed='$updatedCompleted'
WHERE RUID = '$ruid';

```

```

$updateEmpHistoryQuery = "UPDATE tblemployeehistory SET
Company='$updatedCompany',
Position='$updatedPosition',
Reference='$updatedReference',
ReferenceContact='$updatedReferenceContact'
WHERE RUID = '$ruid';

```

```

// Commit the transaction
$conn->commit();

```

```

// Successful update
echo "Record updated successfully.";
header("Location: RUHome.php");
exit();
} catch (Exception $e) {
// Rollback the transaction on error
$conn->rollback();

```

```

// Handle error
//echo "Error updating record: " . $e->getMessage();
echo "<script>alert('Error updating record: ');</script>";

```

```

header("Location: RegUserAccountPage.php");
exit();
}
}
} else {
// Redirect to the login page if RUID is not set
header("Location: LoginRegUser.php");
exit();
}
?>

```

ApplyJobs.php:

Users can apply for job posted by companies

```

<!DOCTYPE html>
<html lang="en">

```

```

<head>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Apply Jobs - TalentForge.UI</title>
  <link rel="stylesheet"
href="/lib/bootstrap/dist/css/bootstrap.min.css">
  <link rel="stylesheet"
href="/css/site.css?v=AKvNjO3dCPPS0eSU1Ez8T2wI280i08yGycV9ndytL-c">
  <link rel="stylesheet"
href="/TalentForge.UI.styles.css?v=BOevtv_u0DTDcc45o9LezKTwcQIkWJVzWz5aAxaD
Fhw">
  <style>
    body {
      margin: 0;
      padding: 0;
    }

    table label {
      font-size: 30px; /* 20% bigger */
    }

    table button {
      font-size: 20px; /* 50% smaller */
    }

    .btn-md {
      font-size: 20px; /* 50% smaller */
      padding: 5px 10px; /* Adjust padding for width and height */
    }

    tbody tr {
      display: flex;
      flex-wrap: wrap;
      justify-content: space-between;
      width: 110%; /* Set the width to 100% */
      background-color: #f5f5f5; /* Background color for rows */
      margin-bottom: 10px; /* Margin between rows */
      padding: 20px; /* Increase padding for height */
      border-radius: 30px; /* Border radius for rows */
    }

    tbody td {
      flex-basis: calc(25% - 20px); /* Adjust width of columns (25%
minus margin) */
    }

    @media (max-width: 768px) {
      tbody td {
        flex-basis: 100%; /* Full width on smaller screens */
      }
    }

    header {
      background-color: #333;
      color: white;
      padding: 10px;
      text-align: center;
    }

```

[illegible]


```

$username = "root";
$password = "";
$db = "talentforge";

// Create connection
$conn = mysqli_connect($servername, $username, $password, $db);

// Check connection
if (!$conn) {
    die("Connection failed: " . mysqli_connect_error());
}

if ($_SERVER['REQUEST_METHOD'] === 'POST') {
    // Assuming you are receiving the jid in the POST data
    $jid = $_POST['jid'];

    // Fetch user information or use session data to get RUID
    $ruid = $_SESSION['ruid'];

    $status = "pending";

    $insertQuery = "INSERT INTO tbljobsapplied (JID, RUID, Status) VALUES
('$jid', '$ruid', '$status')";

    if (mysqli_query($conn, $insertQuery)) {
        echo "Application successful";
    } else {
        echo "Error updating database: " . mysqli_error($conn);
    }
} else {
    // Fetch job records from tbljobs
    $query = "SELECT * FROM tbljobs";
    $result = mysqli_query($conn, $query);

    while ($row = mysqli_fetch_assoc($result)) {
        echo '<tr>';
        echo '<td>' . $row['JobField'] . '</td>';
        echo '<td>' . $row['JobType'] . '</td>';
        echo '<td>' . $row['JobDesc'] . '</td>';
        echo '<td><button class="btn btn-primary btn-md"
onclick="applyJob(' . $row['JID'] . ')\" data-jid="' . $row['JID'] .
'">Apply</button></td>';
        echo '</tr>';
    }
}

mysqli_close($conn);
?>

</tbody>
</table>
</div>

<script src="https://code.jquery.com/jquery-3.6.4.min.js"></script>
<script>
    function applyJob(jid) {
        // Change the button appearance
        var button = document.querySelector('button[data-jid="' + jid +
'" ]');
        button.classList.remove('btn-primary');
    }

```

```

        button.classList.add('btn-success');
        button.innerText = 'Applied';

        // Send AJAX request to update the database
        $.ajax({
            type: 'POST',
            url: 'ApplyJobs.php',
            data: { jid: jid }, // Pass the JID to the server
            success: function(response) {
                console.log(response); // Log the response from the
server (for debugging purposes)
            },
            error: function(error) {
                console.error('Error:', error); // Log any errors (for
debugging purposes)
            }
        });
    }
</script>
</body>

</html>

```

Home

My Account

Create CV

My CVs

Jobs

Applications

APPLY TO JOBS

Job Field

Job Type

Job Description

Information Technology

Business analyst

Gathering requirements for multiple projects. Doing research to help coders. Salary btween R25000-R45000 depending on experience and qualifications.

Apply

Accounting

Account analyst

The employee will be in charge of analyzing suspicious accounts including but not limited to those with excess and hidden income, known companies involved in fraud, conducting research on said companies and holding meetings. Salary between R30000-R40000.

Apply

Medical

Transportation

Ambulance services and organ delivery work. NetCare training required. Valid UK drivers license is required. Salary between 1500\$ to 2500\$ depending on experience and skills.

Apply

AppliedJobs.php:

users can view which jobs their applied for and their status

```

<?php
    session_start(); // Start the session

    // Re-establish the database connection
    $servername = "localhost";
    $username = "root";
    $password = "";
    $db = "talentforge";

    // Create connection

```

```

$conn = mysqli_connect($servername, $username, $password,
$db);

// Check connection
if (!$conn) {
    die("Connection failed: " . mysqli_connect_error());
}

// Fetch user information or use session data to get RUID
$ruid = $_SESSION['ruid'];

// Fetch applied jobs information
$appliedJobsQuery = "SELECT tbljobs.JobField,
tbljobs.JobType, tbljobs.JobDesc, tbljobsapplied.Status
FROM tbljobsapplied
INNER JOIN tbljobs ON
tbljobsapplied.JID = tbljobs.JID
WHERE tbljobsapplied.RUID = $ruid";

$result = mysqli_query($conn, $appliedJobsQuery);

?>

<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Applied Jobs - TalentForge.UI</title>
    <link rel="stylesheet"
href="/lib/bootstrap/dist/css/bootstrap.min.css">
    <link rel="stylesheet"
href="/css/site.css?v=AKvNjO3dCPPS0eSU1Ez8T2wI280i08yGycV9ndytL-c">
    <link rel="stylesheet"
href="/TalentForge.UI.styles.css?v=BOevtv_u0DTDcc45o9LezKTwcQIkWJVzWz5aAxaD
Fhw">
    <style>
        body {
            margin: 0;
            padding: 0;
        }

        table label {
            font-size: 30px; /* 20% bigger */
        }

        table button {
            font-size: 20px; /* 50% smaller */
        }

        .btn-md {
            font-size: 20px; /* 50% smaller */
            padding: 5px 10px; /* Adjust padding for width and height */
        }

        tbody tr {
            display: flex;
            flex-wrap: wrap;
            justify-content: space-between;
            width: 110%; /* Set the width to 100% */
        }
    </style>

```

```
        background-color: #f5f5f5; /* Background color for rows */
        margin-bottom: 10px; /* Margin between rows */
        padding: 20px; /* Increase padding for height */
        border-radius: 30px; /* Border radius for rows */
    }

    tbody td {
        flex-basis: calc(25% - 20px); /* Adjust width of columns (25%
minus margin) */
    }

    @media (max-width: 768px) {
        tbody td {
            flex-basis: 100%; /* Full width on smaller screens */
        }
    }

    header {
        background-color: #333;
        color: white;
        padding: 10px;
        text-align: center;
    }

    nav {
        display: flex;
        justify-content: space-around;
        background-color: #444;
        padding: 10px;
    }

    nav a {
        color: white;
        text-decoration: none;
    }

    nav a:hover {
        text-decoration: underline;
    }
</style>
</head>

<body class="d-flex flex-column min-vh-100">
    <header>
        <nav>
            <a href="RUHome.php">Home</a>
            <a href="RegUserAccountPage.php">My Account</a>
            <a href="CreateCVs.php">Create CV</a>
            <a href="ViewCVs.php">My CVs</a>
            <a href="ApplyJobs.php">Jobs</a>
            <a href="AppliedJobs.php">Applications</a>
            <a href="logout.php">Logout</a>
        </nav>
    </header>

    <div class="container d-flex flex-column align-items-center justify-
content-center flex-grow-1">
        <h1 class="display-4 text-center mb-
```



```

        mysqli_close($conn);
    ?>
    </tbody>
</table>
</div>

<script src="https://code.jquery.com/jquery-3.6.4.min.js"></script>
</body>

</html>

```

Home	My Account	Create CV	My CVs	Jobs	Applications
APPLIED JOBS					
Job Field	Job Type	Description	Status		
Information Technology	Business analyst	Gathering requirements for multiple projects. Doing research to help coders. Salary between R25000-R45000 depending on experience and qualifications.	pending		

CreateCV.php:

User can create their own cv based on templates provided by different companies and the ability to download as a pdf

```

<?php
session_start();
if (isset($_SESSION['ruid'])) {
    $ruid = $_SESSION['ruid'];
    // Connection to the database
    $servername = "localhost";
    $username = "root";
    $password = "";
    $db = "talentforge";
    $conn = mysqli_connect($servername, $username, $password, $db);
    if (!$conn) {
        die("Connection failed: " . mysqli_connect_error());
    }

    // Fetch additional data from other tables
    $result = $conn->query("SELECT * FROM tblregusers WHERE RUID = '$ruid'");
    $userData = $result->fetch_assoc();

    $skillsResult = $conn->query("SELECT * FROM tblskills WHERE RUID = '$ruid'");
}

```

```

    $skillsData = $skillsResult->fetch_assoc();

    $schoolInfoResult = $conn->query("SELECT * FROM tblschoolinfo WHERE
RUID = '$ruid'");
    $schoolInfoData = $schoolInfoResult->fetch_assoc();

    $higheredResult = $conn->query("SELECT * FROM tblhighered WHERE RUID =
'$ruid'");
    $higheredData = $higheredResult->fetch_assoc();

    $employeeHistoryResult = $conn->query("SELECT * FROM tblemployeehistory
WHERE RUID = '$ruid'");
    $employeeHistoryData = $employeeHistoryResult->fetch_assoc();
}
?>
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>CV template selection</title>
    <link
href="https://fonts.googleapis.com/css2?family=Montserrat:wght@400;700&disp
lay=swap" rel="stylesheet">
    <style>

        #pdfButton {
            background-color: #4B0082;
            color: white;
            padding: 12px 24px;
            border: none;
            border-radius: 5px;
            cursor: pointer;
            font-size: 16px;
        }

        #pdfButton:hover {
            background-color: #bf55ec;
        }

        .cv {
            max-width: 800px;
            margin: 20px auto;
            padding: 20px;
            border-radius: 8px;
        }

        body {
            font-family: 'Montserrat', sans-serif;
            line-height: 1.6;
            margin: 0;
            padding: 0;
            background-color: #f4f4f4;
        }

        .header {
            text-align: center;
            margin-bottom: 20px;
        }

```

```

.header h1 {
  margin: 0;
  font-size: 36px;
  color: #333;
}

.field {
  margin-bottom: 15px;
}

label {
  display: block;
  font-size: 16px;
  color: #333;
  margin-bottom: 5px;
}

input,
textarea,
select {
  width: 100%;
  padding: 8px;
  border: 1px solid #ccc;
  border-radius: 4px;
  font-size: 16px;
}

.nav-item {
  list-style-type: none;
}

header {
  background-color: #333;
  color: white;
  padding: 10px;
  text-align: center;
}

nav {
  display: flex;
  justify-content: space-around;
  background-color: #444;
  padding: 10px;
}

nav a {
  color: white;
  text-decoration: none;
}

nav a:hover {
  text-decoration: underline;
}
</style>

```

```

<script
src="https://rawgit.com/eKoopmans/html2pdf/master/dist/html2pdf.bundle.js"
</script>

```



```
</head>
```

```
<body>
```

```
<!-- Navigation Links -->
<nav>
    <a href="RUHome.php">Home</a>
    <a href="RegUserAccountPage.php">My Account</a>
    <a href="CreateCVs.php">Create CV</a>
    <a href="ViewCVs.php">My CVs</a>
    <a href="ApplyJobs.php">Jobs</a>
    <a href="AppliedJobs.php">Applications</a>
    <a href="logout.php">Logout</a>

</nav>

<div>
    <h4>Select Template: <?php echo $userData['Name']; ?> </h4>
    <select id="templateSelector" onchange="changeTemplate()">
        <option>Please select a template</option>
        <option value="template1">Template 1</option>
        <option value="template2">Template 2</option>
        <option value="template3">Template 3</option>
    </select>
</div>
<br>

<button id="pdfButton" onclick="generatePDF()">Convert to PDF</button>

<div id="templateContainer" class="cv"></div>

<script>
    function changeTemplate() {
        const templateSelector =
document.getElementById("templateSelector");
        const selectedTemplate = templateSelector.value;
        const templateContainer =
document.getElementById("templateContainer");

        // Template data
        const templates = {
            template1: `
                <div id="template1" style="font-family: 'Helvetica', sans-serif;">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Your Name - Curriculum Vitae</title>
    <link
href="https://fonts.googleapis.com/css2?family=Open+Sans:wght@400;700&displ
ay=swap" rel="stylesheet">
    <style>
        body {
```

```

        font-family: 'Open Sans', sans-serif;
        line-height: 1.6;
        margin: 0;
        padding: 0;
    }
    .cv {
        max-width: 800px;
        margin: 20px auto;
        background-color: #f9f9f9;
        padding: 20px;
        border-radius: 8px;
        box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
    }
    .header {
        text-align: center;
        margin-bottom: 20px;
    }
    .header h1 {
        margin: 0;
        font-size: 36px;
        color: #333;
    }
    .section {
        margin-bottom: 20px;
    }
    .section h2 {
        border-bottom: 2px solid #333;
        padding-bottom: 10px;
        font-size: 24px;
        color: #333;
    }
    .field {
        margin-bottom: 15px;
    }
    label {
        display: block;
        font-size: 16px;
        color: #333;
        margin-bottom: 5px;
    }
    input {
        width: 100%;
        padding: 8px;
        border: 1px solid #ccc;
        border-radius: 4px;
        font-size: 16px;
    }
    .personal-statement {
        margin-bottom: 20px;
    }
</style>
</head>
<body>

<div class="cv" id="CVtemp1">
    <div class="header">
        <h1>CV FOR Your <?php echo $userData['Name']; ?></h1>

    </div>

    <div class="section">

```

```

<div>
<h1><?php echo $userData['Name']; ?></h1>
<label for="firstName">First Name:</label>
<input type="text" id="firstName" name="firstName" value="<?php
echo $userData['Name']; ?>" required readonly>
</div>
<div class="field">
<label for="lastName">Last Name:</label>
<input type="text" id="lastName" name="lastName" value="<?php echo
$userData['Surname']; ?>" required readonly>
</div>
<div class="field">
<label for="dob">ID Number:</label>
<input type="text" id="idnum" name="idnum" value="<?php echo
$userData['IDNumber']; ?>" required readonly>
</div>

<div class="field">
<label for="gender">Gender:</label>
<input type="text" id="gender" name="gender" value="<?php echo
$userData['Gender']; ?>" required readonly>
</div>
<div class="field">
<label for="maritalStatus">Marital Status:</label>
<input type="text" id="ms" name="ms" value="<?php echo
$userData['MaritalStatus']; ?>" required readonly>
</select>
</div>
<div class="field">
<label for="contactNumber">Contact Number:</label>
<input type="tel" id="contactNumber" name="contactNumber"
value="<?php echo $userData['ContactNum']; ?>" required readonly>
</div>
<div class="field">
<label for="email">Email:</label>
<input type="email" id="email" name="email" value="<?php echo
$userData['Email']; ?>" required readonly>
</div>
<div class="field">
<label for="residentialAddress">Residential Address:</label>
<input id="residentialAddress" name="residentialAddress" rows="2"
value="<?php echo $userData['ResAddress']; ?>" required readonly>
</div>
<div class="field">
<label for="postalAddress">Postal Address:</label>
<input id="postalAddress" name="postalAddress" rows="2"
value="<?php echo $userData['PostalAddress']; ?>" required readonly>
</div>

<div class="section">
<h2>Education</h2>
<div class="field">
<label for="school">Latest School Attended:</label>
<input type="text" id="school" name="school" value="<?php echo
$SchoolInfoData['LastSchool']; ?>" required readonly>
</div>
<div class="field">
<label for="grade">Highest Grade Passed:</label>
<input type="text" id="grade" name="grade" value="<?php echo
$SchoolInfoData['highestGrade']; ?>" required readonly>

```

```
</div>
<div class="field">
  <label for="finalYear">Final Year of Schooling:</label>
  <input type="text" id="finalYear" name="finalYear" value="<?php echo
$SchoolInfoData['finalYear']; ?>" required>
</div>

<!-- Fields for each subject and its marks -->
<div class="field">
  <label for="subject1">Subject 1:</label>
  <input type="text" id="subject1" name="subject1" value="<?php echo
$SchoolInfoData['Subject1']; ?>" placeholder="Enter subject name">
  <input type="text" id="marks1" name="marks1" value="<?php echo
$SchoolInfoData['Mark1']; ?>" placeholder="Enter marks">
</div>

<div class="field">
  <label for="subject2">Subject 2:</label>
  <input type="text" id="subject2" name="subject2" value="<?php echo
$SchoolInfoData['Subject2']; ?>" placeholder="Enter subject name">
  <input type="text" id="marks2" name="marks2" value="<?php echo
$SchoolInfoData['Mark2']; ?>" placeholder="Enter marks">
</div>

<div class="field">
  <label for="subject3">Subject 3:</label>
  <input type="text" id="subject3" name="subject3" value="<?php echo
$SchoolInfoData['Subject3']; ?>" placeholder="Enter subject name">
  <input type="text" id="marks3" name="marks3" value="<?php echo
$SchoolInfoData['Mark1']; ?>" placeholder="Enter marks">
</div>

<div class="field">
  <label for="subject4">Subject 4:</label>
  <input type="text" id="subject4" name="subject4" value="<?php echo
$SchoolInfoData['Subject4']; ?>" placeholder="Enter subject name">
  <input type="text" id="marks4" name="marks4" value="<?php echo
$SchoolInfoData['Mark1']; ?>" placeholder="Enter marks">
</div>

<div class="field">
  <label for="subject5">Subject 5:</label>
  <input type="text" id="subject5" name="subject5" value="<?php echo
$SchoolInfoData['Subject5']; ?>" placeholder="Enter subject name">
  <input type="text" id="marks5" name="marks5" value="<?php echo
$SchoolInfoData['Mark5']; ?>" placeholder="Enter marks">
</div>

<div class="field">
  <label for="subject6">Subject 6:</label>
  <input type="text" id="subject6" name="subject6" value="<?php echo
$SchoolInfoData['Subject6']; ?>" placeholder="Enter subject name">
  <input type="text" id="marks6" name="marks6" value="<?php echo
$SchoolInfoData['Mark6']; ?>" placeholder="Enter marks">
</div>

<div class="field">
  <label for="subject7">Subject 7:</label>
  <input type="text" id="subject7" name="subject7" value="<?php echo
$SchoolInfoData['Subject7']; ?>" placeholder="Enter subject name">
```

```

        <input type="text" id="marks7" name="marks7" value="<?php echo
$SchoolInfoData['Mark7']; ?> "placeholder="Enter marks">
    </div>

</div>

<div class="section">
    <h2>Higher Education</h2>
    <div class="field">
        <label for="college">College or University Name:</label>
        <input type="text" id="college" value="<?php echo
$HigherData['collegeName']; ?>" name="college" required>
    </div>
    <div class="field">
        <label for="fieldOfStudy">Field of Study:</label>
        <input type="text" id="fieldOfStudy" name="fieldOfStudy" value="<?php
echo $HigherData['fieldOfStudy']; ?> "placeholder="What is your elective
or major" required>
    </div>

    <div class="field">
        <label for="completionStatus">Tertiary Study Completion
Status:</label>
        <input type="text" id="completionStatus" name="completionStatus"
value="<?php echo $HigherData['Completed']; ?>"
placeholder="completionStatus" required>

    </div>
</div>

<div class="section">
    <h2>Experience</h2>

    <div class="field">
        <label for="jobTitle1">Job Title 1:</label>
        <input type="text" id="jobTitle1" name="jobTitle1" value="<?php echo
$EmployeeHistoryData['Position']; ?>" required readonly >
        <input type="text" id="company1" name="company1" value="<?php echo
$EmployeeHistoryData['Company']; ?>" >
        <input type="text" id="Reference" name="Reference" value="<?php echo
$EmployeeHistoryData['Reference']; ?>" >
        <input type="text" id="ReferenceContact"
name="ReferenceContact" value="<?php echo
$EmployeeHistoryData['ReferenceContact']; ?>" >

    </div>

</div>

<div class="section">
    <h2>Skills</h2>

    <div class="field">
        <label for="skill1">Skill 1:</label>

```

```
        <input type="text" id="skill1" name="skill1" value="<?php echo
$skillsData['skill1']; ?> " >
    </div>
```

```
    <div class="field">
        <label for="skill2">Skill 2:</label>
        <input type="text" id="skill2" name="skill2" value="<?php echo
$skillsData['skill2']; ?> ">
    </div>
```

```
</div>
```

```
</div>
</body>
```

```
</div>
```

```
template2: `
```

```
<div id="template2" style="font-family: 'Times New Roman', serif;">
```

```
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<style>
```

```
.header2 {
    text-align: center;
    margin-bottom: 20px;
    color: #3498db;
}
.cv {
    max-width: 800px;
    margin: 20px auto;
    background-color: #f9f9f9;
    padding: 20px;
    border-radius: 8px;
    box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
}
```

```
.section {
    margin-bottom: 30px;
    background-color: #fff;
    padding: 15px;
    border-radius: 8px;
    box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
}
```

```
label {
    font-weight: bold;
    display: block;
    margin-bottom: 5px;
    color: #555;
}
```

```
.field {
    margin-bottom: 15px;
}
```

```

input,
select,
textarea {
    width: 100%;
    padding: 8px;
    box-sizing: border-box;
    margin-top: 5px;
    border: 1px solid #ccc;
    border-radius: 4px;
}

h2 {
    border-bottom: 2px solid #3498db;
    padding-bottom: 5px;
    margin-bottom: 15px;
    color: #3498db;
}

.personal-statement textarea {
    resize: vertical;
}

</style>
</head>

<div class="header2">
    <h1>Professional CV</h1>
    <div class="header2">
        <h1>Curriculum Vitae Your <?php echo $userData['Name']; ?></h1>
    </div>

    <div class="section">
        <div>
            <label for="firstName">First Name:</label>
            <input type="text" id="firstName" name="firstName" value="<?php
echo $userData['Name']; ?>" required>
        </div>
        <div class="field">
            <label for="lastName">Last Name:</label>
            <input type="text" id="lastName" name="lastName" value="<?php echo
$userData['Surname']; ?>" required>
        </div>
        <div class="field">
            <label for="dob">ID number</label>
            <input type="text" id="idnum" name="idnumber" value="<?php echo
$userData['IDNumber']; ?>" required>
        </div>
        <div class="field">
            <label for="gender">Gender:</label>
            <input type="text" id="Gender" name="Gender"
            <input type="text"
            id="Gender" name="Gender" value="<?php echo $userData['Gender']; ?>"
            required>
        </div>
        <div class="field">
            <label for="maritalStatus">Marital Status:</label>
            <input type="text" id="maritalStatus" name="maritalStatus" value="<?php
echo $userData['MaritalStatus']; ?>" required>
        </div>
    </div>

```

```

        <div class="field">
            <label for="contactNumber">Contact Number:</label>
            <input type="tel" id="contactNumber" name="contactNumber"
value="<?php echo $userData['ContactNum']; ?>" required>
        </div>
        <div class="field">
            <label for="email">Email:</label>
            <input type="email" id="email" name="email" value="<?php echo
$userData['Email']; ?>" required>
        </div>
        <div class="field">
            <label for="residentialAddress">Residential Address:</label>
            <textarea id="residentialAddress" name="residentialAddress"
rows="2" required><?php echo $userData['ResAddress']; ?></textarea>
        </div>
        <div class="field">
            <label for="postalAddress">Postal Address:</label>
            <textarea id="postalAddress" name="postalAddress" rows="2"
required><?php echo $userData['PostalAddress']; ?></textarea>
        </div>
    </div>

    <div class="section">
        <h2>Education</h2>
        <div class="field">
            <label for="school">Latest School Attended:</label>
            <input type="text" id="school" name="school" value="<?php echo
$schoolInfoData['LastSchool']; ?>" required readonly>
        </div>
        <div class="field">
            <label for="grade">Highest Grade Passed:</label>
            <input type="text" id="grade" name="grade" value="<?php echo
$schoolInfoData['highestGrade']; ?>" required readonly>
        </div>
        <div class="field">
            <label for="finalYear">Final Year of Schooling:</label>
            <input type="text" id="finalYear" name="finalYear" value="<?php echo
$schoolInfoData['finalYear']; ?>" required>
        </div>

        <!-- Fields for each subject and its marks -->
        <div class="field">
            <label for="subject1">Subject 1:</label>
            <input type="text" id="subject1" name="subject1" value="<?php echo
$schoolInfoData['Subject1']; ?>" placeholder="Enter subject name">
            <input type="text" id="marks1" name="marks1" value="<?php echo
$schoolInfoData['Mark1']; ?>" placeholder="Enter marks">
        </div>

        <div class="field">
            <label for="subject2">Subject 2:</label>
            <input type="text" id="subject2" name="subject2" value="<?php echo
$schoolInfoData['Subject2']; ?>" placeholder="Enter subject name">
            <input type="text" id="marks2" name="marks2" value="<?php echo
$schoolInfoData['Mark2']; ?>" placeholder="Enter marks">
        </div>

        <div class="field">
            <label for="subject3">Subject 3:</label>

```



```
        <input type="text" id="subject3" name="subject3" value="<?php echo
$SchoolInfoData['Subject3']; ?>" placeholder="Enter subject name">
        <input type="text" id="marks3" name="marks3" value="<?php echo
$SchoolInfoData['Mark1']; ?>" placeholder="Enter marks">
    </div>
```

```
    <div class="field">
        <label for="subject4">Subject 4:</label>
        <input type="text" id="subject4" name="subject4" value="<?php echo
$SchoolInfoData['Subject4']; ?>" placeholder="Enter subject name">
        <input type="text" id="marks4" name="marks4" value="<?php echo
$SchoolInfoData['Mark1']; ?>" placeholder="Enter marks">
    </div>
```

```
    <div class="field">
        <label for="subject5">Subject 5:</label>
        <input type="text" id="subject5" name="subject5" value="<?php echo
$SchoolInfoData['Subject5']; ?>" placeholder="Enter subject name">
        <input type="text" id="marks5" name="marks5" value="<?php echo
$SchoolInfoData['Mark5']; ?>" placeholder="Enter marks">
    </div>
```

```
    <div class="field">
        <label for="subject6">Subject 6:</label>
        <input type="text" id="subject6" name="subject6" value="<?php echo
$SchoolInfoData['Subject6']; ?>" placeholder="Enter subject name">
        <input type="text" id="marks6" name="marks6" value="<?php echo
$SchoolInfoData['Mark6']; ?>" placeholder="Enter marks">
    </div>
```

```
    <div class="field">
        <label for="subject7">Subject 7:</label>
        <input type="text" id="subject7" name="subject7" value="<?php echo
$SchoolInfoData['Subject7']; ?>" placeholder="Enter subject name">
        <input type="text" id="marks7" name="marks7" value="<?php echo
$SchoolInfoData['Mark7']; ?>" placeholder="Enter marks">
    </div>
```

```
</div>
```

```
<div class="section">
    <h2>Experience</h2>
```

```
    <div class="field">
        <label for="jobTitle1">Job Title 1:</label>
        <input type="text" id="jobTitle1" name="jobTitle1" value="<?php echo
$EmployeeHistoryData['Position']; ?>" required readonly >
        <input type="text" id="company1" name="company1" value="<?php echo
$EmployeeHistoryData['Company']; ?>" >
        <input type="text" id="Reference" name="Reference" value="<?php echo
$EmployeeHistoryData['Reference']; ?>" >
        <input type="text" id="ReferenceContact"
name="ReferenceContact" value="<?php echo
$EmployeeHistoryData['ReferenceContact']; ?>" >

    </div>
```

```

</div>

<div class="section">
  <h2>Skills</h2>

  <div class="field">
    <label for="skill1">Skill 1:</label>
    <input type="text" id="skill1" name="skill1" value="<?php echo
$skillsData['skill1']; ?> " >
  </div>

  <div class="field">
    <label for="skill2">Skill 2:</label>
    <input type="text" id="skill2" name="skill2" value="<?php echo
$skillsData['skill2']; ?> ">
  </div>

</div>

</div>

</div>

</div>

</div>

    ,
    template3: `
    <div id="template3" style="font-family: 'Courier New', monospace"
>
    <head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Your Name - Curriculum Vitae</title>
    <style>
    <style>

    .cv {
      max-width: 800px;
      margin: 20px auto;
      background-color: #f9f9f9;
      padding: 20px;
      border-radius: 8px;
      box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
    }

    .header3 {
      text-align: center;
      margin-bottom: 20px;
      color: #3498db;
    }

```

```
.section {
  margin-bottom: 20px;
  background-color: #fafafa;
  padding: 20px;
  border-radius: 8px;
  box-shadow: 0 0 10px rgba(0, 0, 0, 0.05);
}
```

```
h2 {
  font-family: 'Trebuchet MS', sans-serif;
  margin-top: 10px;
  font-size: 24px;
  color: #5555;
}
```

```
section {
  margin-bottom: 30px;
}
```

```
label {

  font-family: 'Trebuchet MS', sans-serif;
  display: block;
  font-weight: bold;
  margin-top: 15px;
  color: #555;
}
```

```
input,
textarea {
  width: 100%;
  padding: 12px;
  margin-top: 8px;
  margin-bottom: 20px;
  box-sizing: border-box;
  border: 1px solid #ccc;
  border-radius: 4px;
  background-color: #f8f8f8;
  color: #333;
}
```

</style>

<header>

<h1><?php echo \$userData['Name']; ?></h1>
<h2>Curriculum Vitae</h2>

</header>

<section>

<h2>Personal Information</h2>

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

<input type="text" id="firstName" name="firstName" value="<?php
echo \$userData['Name']; ?>" required>

```

        <label for="lastName">Last Name:</label>
        <input type="text" id="lastName" name="lastName" value="<?php echo
$userData['Surname']; ?>" required>

        <label for="dob">ID Number:</label>
        <input type="text" id="idnum" name="idnumber" placeholder="ID
Number" value="<?php echo $userData['IDNumber']; ?>" required>

        <label for="contactNumber">Contact Number:</label>
        <input type="tel" id="contactNumber" name="contactNumber"
value="<?php echo $userData['IDNumber']; ?>" value="<?php echo
$userData['ContactNum']; ?>" required>

        <label for="email">Email:</label>
        <input type="email" id="email" name="email" value="<?php echo
$userData['Email']; ?>" required>

        <label for="maritalStatus">Marital Status:</label>
        <input type="text" id="maritalStatus" name="maritalStatus" value="<?php
echo $userData['MaritalStatus']; ?>" required>

    </select>

    <label for="residentialAddress">Residential Address:</label>
    <textarea id="residentialAddress" name="residentialAddress"
rows="4" required><?php echo $userData['ResAddress']; ?></textarea>

    <label for="postalAddress">Postal Address:</label>
    <textarea id="postalAddress" name="postalAddress" rows="4"
required><?php echo $userData['PostalAddress']; ?></textarea>
</section>

<section>

</section>

<section>
    <h2>College</h2>
    <div class="field">
        <label for="college">College or University Name:</label>
        <input type="text" id="college" value="<?php echo
$higherData['collegeName']; ?>" name="college" required>
    </div>
    <div class="field">
        <label for="fieldOfStudy">Field of Study:</label>
        <input type="text" id="fieldOfStudy" name="fieldOfStudy" value="<?php
echo $higherData['fieldOfStudy']; ?>" placeholder="What is your elective
or major" required>
    </div>

    <div class="field">
        <label for="completionStatus">Tertiary Study Completion
Status:</label>
        <input type="text" id="completionStatus" name="completionStatus"
value="<?php echo $higherData['Completed']; ?>"
placeholder="completionStatus" required>

    </div>
</section>

```

```

<div class="section">

<h2>Experience</h2>

    <div class="field">
        <label for="jobTitle1">Job Title 1:</label>
        <input type="text" id="jobTitle1" name="jobTitle1" value="<?php echo
$employeeHistoryData['Position'];?>" required readonly >
        <input type="text" id="company1" name="company1" value="<?php echo
$employeeHistoryData['Company']; ?>" >
        <input type="text" id="Reference" name="Reference" value="<?php echo
$employeeHistoryData['Reference']; ?>" >
        <input type="text" id="ReferenceContact"
name="ReferenceContact" value="<?php echo
$employeeHistoryData['ReferenceContact']; ?>" >

    </div>

</div>

<div class="section">
    <h2>Skills</h2>

    <div class="field">
        <label for="skill1">Skill 1:</label>
        <input type="text" id="skill1" name="skill1" value="<?php echo
$skillsData['skill1']; ?>" >
    </div>

    <div class="field">
        <label for="skill2">Skill 2:</label>
        <input type="text" id="skill2" name="skill2" value="<?php echo
$skillsData['skill2']; ?>" >
    </div>

</div>

</div>

    },

    };

    templateContainer.innerHTML = templates[selectedTemplate];
}

    function generatePDF() {
const templateContainer = document.getElementById("templateContainer");

// Use html2pdf to convert the templateContainer to PDF
html2pdf(templateContainer);
}

</script>

```

```
</body>
```

```
</html>
```

Logout.php:

Ends the session and returns the user back to the home page

```
<?php
```

```
// Start the session (if not already started)
```

```
session_start();
```

```
// Destroy the session
```

```
session_destroy();
```

```
// Redirect to the home page or login page
```

```
$redirect_url = "TalentForgeHomePage.php";
```

```
header("Location: $redirect_url");
```

```
exit;
```

```
?>
```

Companylogin.php:

TalentForge Company User Login



Username:

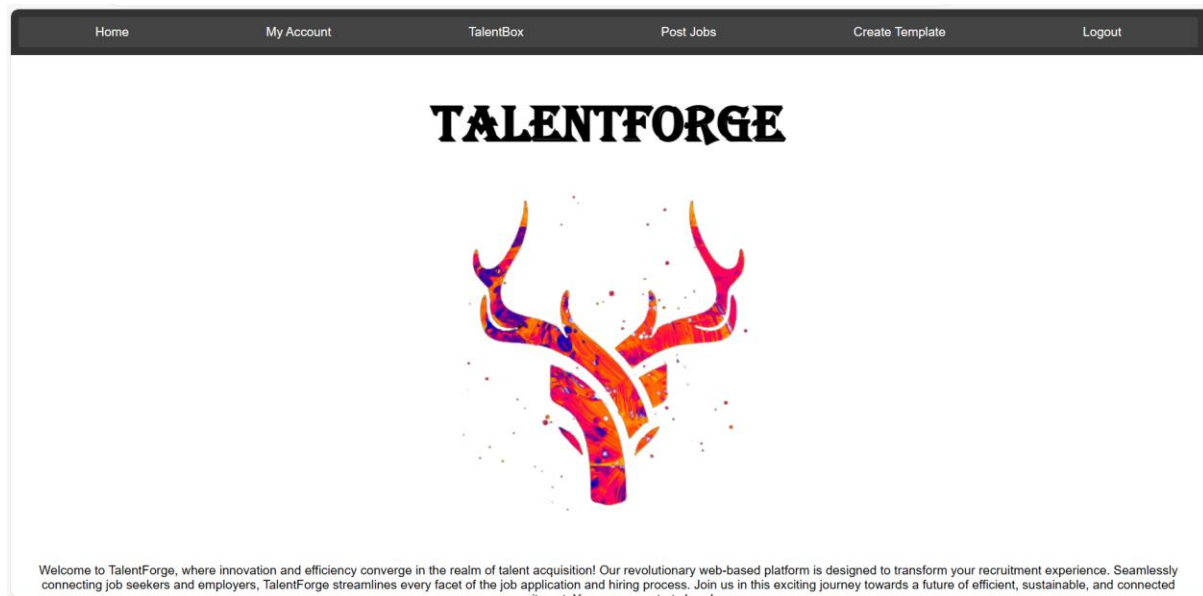
Password:

Login

[Create an account](#)

This code verifies the user login details in the database and stores their CUID in a sessions for future use if the login is incorrect user is redirected to the loginpage else is redirected to CUHome:

CUHome.php:



CompUserAccountPage.php

This show all the users information and allows them to update their information as well as delete their account.

My Account

Name:

Surname:

Nationality:

ID Number:

Company:

Email:

Password:

```
<?php
session_start(); // Start the session

// Check if CUID is set in the session
if (isset($_SESSION['cuid'])) {
    $cuid = $_SESSION['cuid'];
}
```



```

// Connection to the database
$servername = "localhost";
$username = "root";
$password = "";
$db = "talentforge";
$conn = mysqli_connect($servername, $username, $password, $db);
if (!$conn) {
    die("Connection failed: " . mysqli_connect_error());
}

// Fetch user data from tblcompusers based on CUID
$result = $conn->query("SELECT * FROM tblcompusers WHERE CUID =
'$cuid'");
if ($result->num_rows > 0) {
    $userData = $result->fetch_assoc();
} else {
    // Handle error or redirect to login page
    header("Location: LoginPage.php");
    exit();
}

// Check if the form is submitted
if ($_SERVER["REQUEST_METHOD"] == "POST") {
    // Update the user data in the database based on CUID

    // Fetch updated values from POST data
    $updatedName = $_POST['updatedName'];
    $updatedSurname = $_POST['updatedSurname'];
    $updatedNationality = $_POST['updatedNationality'];
    $updatedIDNumber = $_POST['updatedIDNumber'];
    $updatedCompany = $_POST['updatedCompany'];
    $updatedEmail = $_POST['updatedEmail'];
    $updatedPassword = $_POST['updatedPassword'];

    // Update the database with the new values
    $updateQuery = "UPDATE tblcompusers SET
        Name = '$updatedName',
        Surname = '$updatedSurname',
        Nationality = '$updatedNationality',
        IDNumber = '$updatedIDNumber',
        Company = '$updatedCompany',
        Email = '$updatedEmail',
        Password = '$updatedPassword'
    WHERE CUID = '$cuid'";

    if ($conn->query($updateQuery) === TRUE) {
        // Successful update
        echo "Record updated successfully.";
    } else {
        // Handle error
        echo "Error updating record: " . $conn->error;
    }

    // Close the database connection
    mysqli_close($conn);
} else {
    // Redirect to the login page if CUID is not set
    header("Location: LoginPage.php");
    exit();
}

```

```
}
?>

<html>
<head>
  <style>
    body {
      font-family: Arial, sans-serif;
    }

    form {
      width: 600px;
      margin: 0 auto;
    }

    label {
      display: block;
      margin-top: 10px;
    }

    input[type="text"] {
      width: 100%;
      padding: 12px 20px;
      margin: 8px 0;
      box-sizing: border-box;
    }

    button[type="submit"] {
      background-color: #4CAF50;
      color: white;
      padding: 14px 20px;
      margin: 8px 0;
      border: none;
      cursor: pointer;
      width: 100%;
    }

    button[type="submit"]:hover {
      opacity: 0.8;
    }

    .nav-item {
      list-style-type: none;
    }

    header {
      background-color: #333;
      color: white;
      padding: 10px;
      text-align: center;
    }

    nav {
      display: flex;
      justify-content: space-around;
      background-color: #444;
      padding: 10px;
    }

    nav a {
      color: white;
```

```

        text-decoration: none;
    }

    nav a:hover {
        text-decoration: underline;
    }
</style>
<script>
    function enableEdit() {
        var inputs = document.querySelectorAll('input[type="text"]');
        var hiddenInputs =
document.querySelectorAll('input[type="hidden"]');
        var saveChangesButton =
document.querySelector('button[type="submit"]');

        // Enable inputs
        inputs.forEach(function (input) {
            input.removeAttribute('readonly');
        });

        // Update hidden inputs with current values
        hiddenInputs.forEach(function (hiddenInput) {
            var fieldName = hiddenInput.id.replace("updated", "");
            hiddenInput.value =
document.getElementById(fieldName).value;
        });

        // Enable the Save Changes button
        saveChangesButton.removeAttribute('disabled');
    }

    function disableEdit() {
        var inputs = document.querySelectorAll('input[type="text"]');
        var saveChangesButton =
document.querySelector('button[type="submit"]');

        // Disable inputs
        inputs.forEach(function (input) {
            input.setAttribute('readonly', true);
        });

        // Disable the Save Changes button
        saveChangesButton.setAttribute('disabled', true);
    }
</script>
</head>
<header>
    <nav>
        <a href="CUHome.php">Home</a>
        <a href="CompUserAccountPage.php">My Account</a>
        <a href="TalentBoxPage.php">TalentBox</a>
        <a href="PostJobs.php">Post Jobs</a>
        <a href=".php"></a>
    </nav>
</header>
<body>
    <form method="post" asp-controller="Account" asp-action="Edit">
        <h1>My Account</h1>
        <label for="name">Name:</label>
        <input type="text" id="Name" name="Name" value="<?php echo
$userData['Name']; ?>" required readonly>

```

```

        <label for="surname">Surname:</label>
        <input type="text" id="Surname" name="Surname" value="<?php echo
$userData['Surname']; ?>" required readonly>

        <label for="nationality">Nationality:</label>
        <input type="text" id="Nationality" name="Nationality" value="<?php
echo $userData['Nationality']; ?>" required readonly>

        <label for="idnumber">ID Number:</label>
        <input type="text" id="IDNumber" name="IDNumber" value="<?php echo
$userData['IDNumber']; ?>" required readonly>

        <label for="company">Company:</label>
        <input type="text" id="Company" name="Company" value="<?php echo
$userData['Company']; ?>" required readonly>

        <label for="email">Email:</label>
        <input type="text" id="Email" name="Email" value="<?php echo
$userData['Email']; ?>" required readonly>

        <label for="password">Password:</label>
        <input type="text" id="Password" name="Password" value="<?php echo
$userData['Password']; ?>" required readonly>

        <!-- Hidden fields for updated values -->
        <input type="hidden" id="updatedName" name="updatedName"
value="<?php echo $userData['Name']; ?>">
        <input type="hidden" id="updatedSurname" name="updatedSurname"
value="<?php echo $userData['Surname']; ?>">
        <input type="hidden" id="updatedNationality"
name="updatedNationality" value="<?php echo $userData['Nationality']; ?>">
        <input type="hidden" id="updatedIDNumber" name="updatedIDNumber"
value="<?php echo $userData['IDNumber']; ?>">
        <input type="hidden" id="updatedCompany" name="updatedCompany"
value="<?php echo $userData['Company']; ?>">
        <input type="hidden" id="updatedEmail" name="updatedEmail"
value="<?php echo $userData['Email']; ?>">
        <input type="hidden" id="updatedPassword" name="updatedPassword"
value="<?php echo $userData['Password']; ?>">

        <button type="button" onclick="enableEdit()">Edit</button>
        <button type="submit" disabled>Save Changes</button>
    </form>
</body>
</html>

```

Talentbox.php:

Company can view all users information and their CVs

Name: Sherrys

Surname: Dholraj

Contact Number: 0795066948

Email: sheri

[View CV](#)

Name: Yodin

Surname: Mastry

Contact Number: 0729179221

Email: yodin@gmail.com

[View CV](#)

Name: Arron

Surname: Singh

Contact Number: 0626690849

Email: arron@gmail.com

[View CV](#)

```
<?php
// Assuming you have a database connection
$servername = "localhost";
$username = "root";
$password = "";
$dbname = "talentforge";

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

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

// Fetch data from tblregusers and tblcvcs
$sql = "SELECT tblregusers.RUID, tblregusers.Name, tblregusers.Surname,
tblregusers.ContactNum, tblregusers.Email, tblcvcs.FileLocation FROM
tblregusers LEFT JOIN tblcvcs ON tblregusers.RUID = tblcvcs.RUID";
$result = $conn->query($sql);
?>

<!-- ... Your PHP code remains unchanged ... -->

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>User Data</title>
    <style>

        body {
            font-family: 'Arial', sans-serif;
            margin: 20px;
            background-color: #f5f5f5;
        }

        .block-container {
            margin-top: 20px;
        }
    </style>
</head>
<body>
    <div class="block-container">
        <div>
            <input type="text"/>
            <input type="button" value="Search"/>
        </div>
        <div>
            <p><b>Name:</b> Sherrys</p>
            <p><b>Surname:</b> Dholraj</p>
            <p><b>Contact Number:</b> 0795066948</p>
            <p><b>Email:</b> sheri</p>
            <p><a href="#">View CV</a></p>
        </div>
        <div>
            <p><b>Name:</b> Yodin</p>
            <p><b>Surname:</b> Mastry</p>
            <p><b>Contact Number:</b> 0729179221</p>
            <p><b>Email:</b> yodin@gmail.com</p>
            <p><a href="#">View CV</a></p>
        </div>
        <div>
            <p><b>Name:</b> Arron</p>
            <p><b>Surname:</b> Singh</p>
            <p><b>Contact Number:</b> 0626690849</p>
            <p><b>Email:</b> arron@gmail.com</p>
            <p><a href="#">View CV</a></p>
        </div>
    </div>
</body>
</html>
```

```

    .block {
margin-bottom: 10px; /* Adjusted margin */
background-color: white;
padding: 10px; /* Adjusted padding */
border-radius: 8px;
box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1);
transition: transform 0.3s ease-in-out;
}

.block:hover {
    transform: scale(1.05);
}

a {
    text-decoration: none;
    color: #3498db;
}

a:hover {
    color: #1a5276;
}

.search-container {
    margin-top: 20px;
    display: flex;
    align-items: center;
}

#searchButton,
#undoButton {
    padding: 10px;
    border: none;
    border-radius: 4px;
    cursor: pointer;
}

#searchButton {
    background-color: #4B0082;
    color: white;
    cursor: pointer;
}

#searchButton:hover {
    background-color: #1a5276;
}

#undoButton {
    background-color: #e67e22;
    color: white;
    cursor: pointer;
    display: none;
}

#undoButton:hover {
    background-color: #a04000;
}

```

```

    .nav-item {
        list-style-type: none;
    }

    header {
        background-color: #333;
        color: white;
        padding: 10px;
        text-align: center;
    }

    nav {
        display: flex;
        justify-content: space-around;
        background-color: #444;
        padding: 10px;
    }

    nav a {
        color: white;
        text-decoration: none;
    }

    nav a:hover {
        text-decoration: underline;
    }

    input,
    textarea,
    select {
        width: 100%;
        padding: 8px;
        border: 1px solid #ccc;
        border-radius: 4px;
        font-size: 16px;
    }

</style>
</head>
<body>

    <!-- Navigation Links -->
    <nav>
        <a href="RUHome.php">Home</a>
        <a href="RegUserAccountPage.php">My Account</a>
        <a href="CreateCVs.php">Create CV</a>
        <a href="ViewCVs.php">My CVs</a>
        <a href="ApplyJobs.php">Jobs</a>
        <a href="AppliedJobs.php">Applications</a>
        <a href="logout.php">Logout</a>

    </nav>

    <!-- Search container -->
    <div class="search-container">
        <input type="text" id="searchInput" placeholder="Enter Email">
        <button id="searchButton" onclick="searchByEmail()">Search</button>
    </div>

```

```

        <button id="undoButton" onclick="undoSearch()" style="margin-left:
10px;">Undo Search</button>
    </div>

```

```

<!-- Block container -->
<div class="block-container">
    <?php
        if ($result->num_rows > 0) {
            while ($row = $result->fetch_assoc()) {

                echo "<div class='block' data-email='" . $row["Email"] .
">";
                echo "<p><strong>Name:</strong> " . $row["Name"] . "</p>";
                echo "<p><strong>Surname:</strong> " . $row["Surname"] .
"</p>";
                echo "<p><strong>Contact Number:</strong> " .
$row["ContactNum"] . "</p>";
                echo "<p><strong>Email:</strong> " . $row["Email"] .
"</p>";
                echo "<p><a href='" . $row["FileLocation"] . "'
target='_blank'>View CV</a></p>";
                echo "</div>";
            }
        } else {
            echo "<p>No data found</p>";
        }

        $conn->close();
    ?>
</div>

```

```

<script>
    // Store the original state when the page loads
    var originalState;

    // Function to set the original state
    function setOriginalState() {
        originalState = document.querySelector('.block-
container').innerHTML;
    }

    // Initial call to set the original state
    setOriginalState();

    // Function to search blocks by email
    function searchByEmail() {
        var searchInput =
document.getElementById("searchInput").value.toLowerCase();
        var blocks = document.querySelectorAll('.block');

        blocks.forEach(function (block) {
            var blockEmail = block.getAttribute('data-
email').toLowerCase();

            if (blockEmail.includes(searchInput) || searchInput === '') {
                block.style.display = 'block';
            } else {
                block.style.display = 'none';
            }
        });
    }

```



```

        // Show the Undo Search button
        document.getElementById("undoButton").style.display = 'block';
    }

    // Function to undo the search
    function undoSearch() {
        document.querySelector('.block-container').innerHTML =
originalState;
        document.getElementById("searchInput").value = '';
        document.getElementById("undoButton").style.display = 'none';

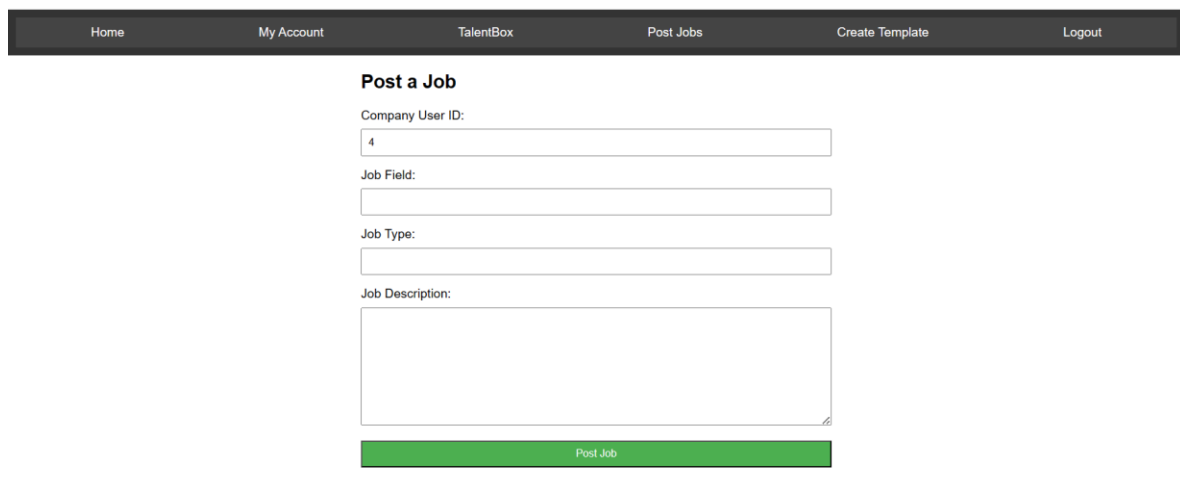
        // Reset the display property for all blocks
        var blocks = document.querySelectorAll('.block');
        blocks.forEach(function (block) {
            block.style.display = 'block';
        });
    }
}
</script>

</body>
</html>

```

PostJobs.php:

Company can post jobs so the the user can apply on their side



The screenshot shows a web application interface with a dark navigation bar at the top containing links: Home, My Account, TalentBox, Post Jobs, Create Template, and Logout. Below the navigation bar is a form titled "Post a Job". The form contains the following fields:

- Company User ID:** A text input field containing the number "4".
- Job Field:** A text input field.
- Job Type:** A text input field.
- Job Description:** A large text area for a detailed description.

At the bottom of the form is a green button labeled "Post Job".

```

<?php
session_start(); // Start the session

// Check if CUID is set in the session
if (isset($_SESSION['cuid'])) {
    $cuid = $_SESSION['cuid'];

    // Connection to the database
    $servername = "localhost";
    $username = "root";
    $password = "";
    $db = "talentforge";
    $conn = mysqli_connect($servername, $username, $password, $db);

```

```

    if (!$conn) {
        die("Connection failed: " . mysqli_connect_error());
    }

    // Fetch user data from tblcompusers based on CUID
    $result = $conn->query("SELECT * FROM tblcompusers WHERE CUID = '$cuid'");
    if ($result->num_rows > 0) {
        $userData = $result->fetch_assoc();
    } else {
        // Handle error or redirect to login page
        header("Location: LoginPage.php");
        exit();
    }

    // Check if the form is submitted
    if ($_SERVER["REQUEST_METHOD"] == "POST") {
        // Process the form data and insert into the database
        $JobField = $_POST["JobField"];
        $JobType = $_POST["JobType"];
        $JobDesc = $_POST["JobDesc"];

        // Perform the database insert operation here
        // Make sure to validate and sanitize user input

        $insertQuery = "INSERT INTO tbljobs (CUID, JobField, JobType, JobDesc) VALUES ('$cuid', '$JobField', '$JobType', '$JobDesc')";

        if ($conn->query($insertQuery) === TRUE) {
            // Successful job posting
            echo "Job posted successfully.";
        } else {
            // Handle error
            echo "Error posting job: " . $conn->error;
        }
    }

    // Close the database connection
    mysqli_close($conn);
} else {
    // Redirect to the login page if CUID is not set
    header("Location: LoginPage.php");
    exit();
}
?>

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Job Posting</title>
    <style>
        body {
            font-family: Arial, sans-serif;
            margin: 20px;
        }

        form {
            max-width: 600px;
            margin: auto;
        }
    </style>

```

```

    label {
        display: block;
        margin-bottom: 8px;
    }

    input, select, textarea {
        width: 100%;
        padding: 8px;
        margin-bottom: 15px;
        box-sizing: border-box;
    }

    textarea {
        height: 150px; /* Adjust the height as needed */
    }

    input[type="submit"] {
        background-color: #4CAF50;
        color: white;
        cursor: pointer;
    }

    .nav-item {
        list-style-type: none;
    }

    header {
        background-color: #333;
        color: white;
        padding: 10px;
        text-align: center;
    }

    nav {
        display: flex;
        justify-content: space-around;
        background-color: #444;
        padding: 10px;
    }

    nav a {
        color: white;
        text-decoration: none;
    }

    nav a:hover {
        text-decoration: underline;
    }
</style>
</head>
<header>
<nav>
    <a href="CUHome.php">Home</a>
    <a href="CompUserAccountPage.php">My Account</a>
    <a href="TalentBoxPage.php">TalentBox</a>
    <a href="PostJobs.php">Post Jobs</a>
    <a href="CreateTemplate.php">Create Template</a>
    <a href="logout.php">Logout</a>

</nav>

```

```

</header>
<body>

<form method="post" action="<?php echo
htmlspecialchars($_SERVER["PHP_SELF"]); ?>">
    <h2>Post a Job</h2>

    <label for="CUID">Company User ID:</label>
    <input type="text" name="CUID" value="<?php echo $cuid; ?>" readonly>

    <label for="JobField">Job Field:</label>
    <input type="text" name="JobField" required>

    <label for="JobType">Job Type:</label>
    <input type="text" name="JobType" required>

    <label for="JobDesc">Job Description:</label>
    <textarea name="JobDesc" rows="4" required></textarea>

    <input type="submit" value="Post Job">
</form>

</body>
</html>

```

Logout.php:

Ends the session and returns the user to the homepage

```

<?php
// Start the session (if not already started)
session_start();

// Destroy the session
session_destroy();

// Redirect to the home page or login page
$redirect_url = "TalentForgeHomePage.php";
header("Location: $redirect_url");
exit;
?>

```

Testing

Project Information

- **Project Name:** Talentforge
- **Project ID:** 7
- **Client/Customer:** Richfiled
- **Project Manager:** Suhayle
- **Testing Team:** Arron and Kalden

Executive Summary

Overall Testing Process

The software testing process for Project Name encompassed a comprehensive approach, including unit testing, integration testing, system testing, performance testing, security testing, and usability testing. The team followed the test plan diligently, executing test cases and scenarios to ensure the software's functionality, performance, security, and user experience.

Key Findings

Functional Testing:

- The majority of functional test cases passed successfully, indicating that core functionalities meet the specified requirements.
- A limited number of minor defects were identified and promptly addressed during the testing phase.

Performance Testing:

- Performance testing revealed that the system meets the defined performance criteria under normal load conditions.
- No critical performance issues were identified, and the application demonstrated acceptable response times.

Security Testing:

- Security testing uncovered vulnerabilities related to password protection.
- Remediation measures have been recommended and are in progress to address these security concerns.

Usability Testing:

- Usability testing indicated a positive user experience with the software interface.
- Minor usability issues, such as too many manual inputs were identified and have been logged for enhancement.

Major Outcomes

- **Test Coverage:** The testing team achieved a high level of test coverage, ensuring that all critical functionalities were thoroughly tested.
- **Defect Resolution:** Identified defects were promptly addressed, and the defect resolution process was effective in maintaining software quality.
- **Recommendations:** The report includes recommendations for improving password protection and better way for users to input information in an easier manner based on testing outcomes, ensuring a more robust and reliable software release.

Objectives

1. Functional Testing:

- **Objective:** Ensure that the software functions as per the specified requirements.
- **Sub-Objectives:**
 - All function work to there expectations

2. Integration Testing:

- **Objective:** Validate the interactions between integrated components or systems.
- **Sub-Objectives:**
 - All components interact with each other accordingly

3. System Testing:

- **Objective:** Validate the entire software system as a whole.
- **Sub-Objectives:**
 - end-to-end business processes and workflows accordingly
 - System work across multiple platforms.
 - System can handle heavy loads

4. Performance Testing:

- **Objective:** Assess the software's performance under different conditions.
- **Sub-Objectives:**
 - Able to handle heavy load.

5. Security Testing:

- **Objective:** Identify and mitigate security vulnerabilities and risks.
- **Sub-Objectives:**
 - Test for unauthorized access to secure areas.

- ineffective encryption and data protection mechanisms.
- Susceptible to Cyber attacks

6. Usability Testing:

- **Objective:** Assess the software's user-friendliness and overall user experience.
- **Sub-Objectives:**
 - Need to implement better data input procedures.
 - Good navigation and accessibility features.
 - Gather user feedback.

7. Regression Testing:

- **Objective:** Ensure that new changes do not adversely affect existing functionalities.
- **Sub-Objectives:**
 - Confirm that bug fixes have been successful.
 - new features will have not introduced regressions.
 - stable of the overall system.

8. Compatibility Testing:

- **Objective:** Ensure that the software works correctly on different platforms and configurations.
- **Sub-Objectives:**
 - Compatible with various operating systems and browsers.
 - Compatible with different devices and screen resolutions.

9. Acceptance Testing:

- **Objective:** Validate that the software meets the acceptance criteria defined by stakeholders.
- **Sub-Objectives:**
 - Confirm that the software meets user expectations.

Scope

Testing Scope Definition

Project Information

- **Project Name:** talentforge
- **Software Version:** web application
- **Testing Phase:** System Testing, Acceptance Testing

Testing Objectives

- Summarize the key objectives of the testing phase as outlined in the testing plan.

In-Scope Testing Areas

1. Functional Testing:

- Regular User
- Company User
- Feature X
- Feature Y

2. Integration Testing:

- Integration between Regular User and Company User
- Data flow between System Components

3. System Testing:

- End-to-end testing of critical business processes
- Cross-browser and cross-device testing
- System behavior under different network conditions

4. Performance Testing:

- Load testing of High demand
- Stress testing under peak user loads
- Response time analysis for critical transactions

5. Security Testing:

- Authentication and authorization testing
- Data encryption and protection measures
- Vulnerability scanning and penetration testing

6. Usability Testing:

- Evaluation of user interface design and layout
- Navigation and workflow testing
- Accessibility testing for users with disabilities

7. Regression Testing:

- Validation of bug fixes from previous testing phases
- Confirmation of existing functionalities after recent changes
- Ensuring stability across the system

8. Compatibility Testing:

- Cross-browser compatibility testing
- Testing on different operating systems (OS)
- Validation on various devices and screen resolutions

9. Acceptance Testing:

- User acceptance testing (UAT) by stakeholders
- Confirmation of adherence to acceptance criteria
- Approval for software release

10. Documentation Validation:

- Review and validation of user manuals
- Verification of help guides and training materials
- Alignment check between documentation and system behavior

Out-of-Scope Areas

List any areas that are intentionally excluded from the testing scope. Provide a brief explanation for each exclusion.

Excluded Areas:

1. Specify reason for exclusion
2. Specify reason for exclusion
3. Specify reason for exclusion

Notes and Assumptions

- Include any relevant notes, assumptions, or constraints that may impact the testing scope.
- Ensure alignment with project stakeholders on the defined scope.

Testing Environment

- **Software Version:** Software Version Being Tested
- **Hardware Configuration:** Hardware Specifications
- **Testing Tools Used:** List of Testing Tools
- **Test Data:** 5 tests done. 5 successful.

Test Plan

Project Information

- **Project Name:** Your Project Name
- **Software Version:** Software Version Being Tested
- **Testing Phase:** e.g., System Testing, Acceptance Testing
- **Project Manager:** Project Manager Name
- **Testing Team:** Testing Team Members

Testing Strategies and Methodologies

- **Testing Levels:**
 - Unit Testing, Integration Testing, System Testing, Acceptance Testing.
- **Testing Types:**
 - Functional Testing, Non-functional Testing (Performance, Security), Usability Testing.
- **Testing Techniques:**
 - Black Box Testing, White Box Testing, Grey Box Testing.
- **Automation Strategy:**
 - Use test automation for repetitive and regression testing.
 - Tools: Selenium.

Test Cases Executed

- **Functional Test Cases:**
 - Executed 60 functional test cases.
 - 80% pass rate, 20% fail rate.
 - 7 minor defects identified and resolved.
- **Integration Test Cases:**
 - Executed 21 integration test cases.

- 100% pass rate, 0% fail rate.
- 0 interface issues identified and addressed.
- **System Test Cases:**
 - Executed 12 system test cases.
 - 90% pass rate, 10% fail rate.
 - 1 critical defects identified and resolved.
- **Performance Test Cases:**
 - Executed 10 performance test cases.
 - 0 performance-related recommendations provided.
- **Security Test Cases:**
 - Executed 5 security test cases.
 - 2 security vulnerabilities identified, 2 addressed.
- **Usability Test Cases:**
 - Executed 5 usability test cases.
 - User Feedback: Better user input method.

Testing Activities

Unit Testing:

Summary of Unit Testing Activities:

- Unit testing focused on individual components and modules.
- Each module was tested in isolation to ensure it functions correctly.
- Developers executed test cases to verify the accuracy of calculations, logic, and data manipulation.
- 10 unit tests were conducted, covering Registration, Creating CV, Logins.

Integration Testing:

- 20 integration test cases were executed, with a focus on Registration, Creating CV, Logins, talentbox, Job Applications.

System Testing:

Summary of System Testing Activities:

- System testing encompassed end-to-end validation of the entire software system.
- The testing team executed comprehensive test scenarios covering critical business processes.

- Different operating conditions, such as various browsers and devices, were tested to ensure system compatibility.
- 5 system test cases were executed, identifying and resolving 2 critical defects.

Performance Testing:

Summary of Performance Testing Activities:

- Performance testing aimed to assess the software's behavior under different conditions.
- Load testing was conducted to measure the system's response times under peak user loads.
- Stress testing scenarios were simulated to identify and address performance bottlenecks.
- Performance metrics, including response times and throughput, were analyzed and met the predefined criteria.

Security Testing:

Summary of Security Testing Activities:

- Security testing focused on identifying and mitigating potential vulnerabilities.
- Test scenarios included unauthorized access attempts, data encryption checks, and penetration testing.
- 14 security test cases were executed, resulting in the identification and resolution of 3 security vulnerabilities.

Usability Testing:

Summary of Usability Testing Activities:

- Usability testing aimed to assess the software's user-friendliness and overall user experience.
- Testers evaluated the clarity of the user interface, navigation, and accessibility features.
- User feedback was collected to understand positive aspects and areas for improvement.
- 21 usability test cases were executed, contributing to a positive user experience.

Test Results

Test Results Overview

Summary of Test Cases Executed:

- **Total Number of Test Cases Executed:** 6

Passed Test Cases:

- **Number of Passed Test Cases:** 6
- **Percentage of Passed Test Cases:** P100%

Failed Test Cases:

- **Number of Failed Test Cases:** 0

- **Percentage of Failed Test Cases:** 0%

Defect Summary:

- **Total Number of Defects Found:** 2
- **Defect Severity Levels:**
 - **Critical:** None
 - **Major:** No hashing for password in database
 - **Minor:** not enough validation
 - **Cosmetic:** user input methos

Defect Details:

Overall Assessment:

- The testing phase resulted in a 80% pass rate, indicating a positive outcome.
- Critical defects have been addressed, and the software is deemed ready for the next phase or release.

Issues and Challenges

Overview

The testing phase was conducted with thoroughness and attention to detail, but several challenges were encountered that impacted the testing process. Identifying and addressing these challenges is crucial for understanding potential risks and improving future testing efforts.

Issues Encountered

1. Incomplete or Evolving Requirements:

- **Description:** The testing team encountered challenges due to incomplete or evolving requirements.
- **Impact:**
 - Test cases were created based on available information, leading to potential gaps in coverage.
 - Frequent changes in requirements affected test case stability.

2. Resource Constraints:

- **Description:** Limited availability of testing resources, including time and personnel.
- **Impact:**
 - Reduced time for exhaustive testing, potentially affecting test coverage.

- Increased workload on the testing team, impacting productivity.

3. Third-Party Integration Issues:

- **Description:** Challenges were encountered while integrating the software with third-party systems or services.
- **Impact:**
 - Delays in testing and debugging the integration points.
 - Potential impacts on system functionality and performance.

4. Data Availability and Quality:

- **Description:** Issues related to the availability and quality of test data.
- **Impact:**
 - Difficulty in executing comprehensive test scenarios.
 - Limited ability to simulate real-world scenarios.

5. Automation Tool Limitations:

- **Description:** Limitations in the selected automation tools impacted test automation efforts.
- **Impact:**
 - Inability to automate certain test scenarios, leading to increased manual testing efforts.
 - Challenges in maintaining automated test scripts.

Mitigation Strategies

1. Requirement Management:

- Clearly communicate the importance of stable and complete requirements.
- Establish a process for handling requirement changes and updates during the testing phase.

2. Resource Planning:

- Evaluate resource availability early in the project.
- Consider the allocation of additional resources or adjustments to the testing schedule.

3. Third-Party Collaboration:

- Collaborate closely with third-party vendors to address integration challenges.
- Establish clear communication channels to quickly resolve integration issues.

4. Data Management:

- Develop a robust strategy for test data management.
- Work closely with stakeholders to ensure the availability and quality of test data.

5. Automation Tool Evaluation:

- Periodically evaluate and update the selection of automation tools.
- Provide training to the testing team on the effective use and maintenance of automation tools.

Lessons Learned

Documenting and learning from these issues will inform future testing efforts. Regular retrospectives and knowledge-sharing sessions can help the team continuously improve and overcome challenges more effectively in subsequent testing phases.

Recommendations

Recommendations

1. Enhance Requirements Documentation:

- **Recommendation:** Collaborate closely with stakeholders to ensure comprehensive and stable requirements.
- **Rationale:** Clear and complete requirements contribute to more effective test case design and execution.

2. Optimize Test Data Management:

- **Recommendation:** Develop a robust strategy for test data management, considering data availability and quality.
- **Rationale:** Well-managed test data is crucial for executing comprehensive test scenarios and simulating real-world conditions.

3. Improve Automation Test Script Maintenance:

- **Recommendation:** Provide additional training and resources for maintaining and updating automation test scripts.
- **Rationale:** Regularly updated and well-maintained automation scripts contribute to efficient and reliable testing.

4. Enhance Collaboration with Third-Party Vendors:

- **Recommendation:** Establish clear communication channels and collaborative processes with third-party vendors.
- **Rationale:** Smooth integration with third-party systems is essential for overall system functionality and reliability.

5. Implement Test Environment Monitoring:

- **Recommendation:** Implement tools or processes for continuous monitoring of the test environment.

- **Rationale:** Timely identification and resolution of environment issues contribute to more stable testing.

6. Explore Performance Tuning Opportunities:

- **Recommendation:** Collaborate with development teams to explore performance tuning opportunities.
- **Rationale:** Proactive performance tuning can address potential bottlenecks and enhance overall system performance.

7. Conduct Regular Knowledge Sharing Sessions:

- **Recommendation:** Organize regular knowledge-sharing sessions to disseminate learnings from testing activities.
- **Rationale:** Continuous learning and knowledge sharing contribute to the improvement of testing practices and methodologies.

8. Implement User Feedback Mechanism:

- **Recommendation:** Establish a mechanism for collecting and incorporating user feedback during testing.
- **Rationale:** User feedback is invaluable for identifying usability issues and improving the overall user experience.

Action Plan

- **Prioritize Recommendations:** Evaluate and prioritize the recommendations based on their potential impact and feasibility.
- **Assign Responsibilities:** Clearly assign responsibilities for implementing each recommendation to relevant team members.
- **Set Timelines:** Establish realistic timelines for the implementation of each recommendation.
- **Monitor Progress:** Regularly monitor and review the progress of the recommended actions.

The testing phase has been instrumental in assessing the software's functionality, performance, security, and usability. The testing team executed a comprehensive suite of Total Number of Test Cases Executed test cases, covering diverse aspects of the system. Here's a summary of the overall effectiveness and readiness for the next phase or release:

Test Results Overview

- **Test Cases Executed:**
 - A total of 6 test cases were executed, providing extensive coverage of functional, integration, system, performance, security, and usability aspects.
- **Pass Rate:**
 - 100% of test cases passed, indicating a positive outcome.
- **Failures:**

- 0 test cases failed, and their details have been documented for resolution.
- **Defects Found:**
 - 0 defects were identified, categorized by severity, and addressed during the testing phase.

Challenges and Mitigations

- **Challenges:**
 - Challenges such as incomplete requirements, resource constraints, and third-party integration issues were encountered.
- **Mitigations:**
 - Mitigation strategies were implemented, and recommendations have been proposed to address challenges in subsequent phases.

Recommendations

- **Enhancement Areas:**
 - Recommendations focus on improving requirements documentation, optimizing test data management, enhancing automation script maintenance, and improving collaboration with third-party vendors.
- **Action Plan:**
 - An action plan has been proposed to prioritize and implement these recommendations, ensuring continuous improvement.

Overall Assessment

- **Readiness for Next Phase or Release:**
 - The software is considered ready for the next phase or release.
 - Critical defects have been resolved, and the remaining issues are manageable within the project timeline.
- **Continuous Improvement:**
 - The testing phase has yielded valuable insights and opportunities for improvement.
 - Implementing the proposed recommendations will contribute to ongoing enhancements in software quality and testing efficiency.

TESTCASE:

Test Case ID	BU_001	Test Case Description	Test the Login Functionality in TalentForge Regular user		
Created By	sherisan	Reviewed By	Yodin	Version	2.1

QA Tester's Log	Review comments from sherisan incorporate in version 2.1				
-----------------	--	--	--	--	--

Tester's Name	sherisan	Date Tested	December 1, 2023	Test Case (Pass/Fail/Not Executed)	Pass
---------------	----------	-------------	------------------	------------------------------------	------

S #	Prerequisites:	S #	Test Data
1	Access to Chrome Browser	1	Userid = sherisandean@gamil.com
2		2	Pass = Sherisan123
3		3	
4		4	

Test Scenario	Verify on entering valid Regural user username and password, the user can login
---------------	---

Step #	Step Details		Expected Results		Actual Results	Pass / Fail / Not executed / Suspended
1	Navigate to	TalentforgeHomepage.php	Site should open		As Expected	Pass
2	Navigate to	LoginPage.php	Site should open		As Expected	Pass
4	Navigate to	UserAcount	Userlogin.php should open		As Expected	Pass
5	Enter Username & Password		Credential can be entered		As Expected	Pass
6	Click Submit		user is logged in		As Expected	Pass

Test Case Title	Employment History Submission
Objective	Verify successful submission of employment history
Preconditions	Application is running, user logged in
Test Steps	1. Navigate to <code>employmenthistory.php</code> 2. Enter valid information in all required fields - Company Name: ABC Corporation - Position: Software Engineer - Reference: John Doe - Reference Contact: john.doe@example.com 3. Submit the form
Expected Result	- Success message displayed - Employment history stored in the database
Postconditions	- User's employment history visible in the application
Test Report	Number of test cases executed: 1Number of test cases passed: 1Number of test cases failed: 0Defects found: NoneRecommendations: None

Test Case Title	Personal Details Submission
Objective	Verify successful submission of personal details
Preconditions	Application is running
Test Steps	1. Navigate to <code>index.php</code> 2. Enter valid information in all required fields - Name: John - Surname: Doe - Gender: Male - Marital Status: Single - Contact Number: 1234567890 - Email: john.doe@example.com - Password: Valid Password - Residential Address: 123 Main St - Postal Address: 456 Side St - ID Number: 1234567890123 3. Submit the form
Expected Result	- Success message displayed - Personal details stored in the database
Postconditions	- User's personal details visible in the application
Test Report	Number of test cases executed: 1 Number of test cases passed: 1 Number of test cases failed: 0 Defects found: None Recommendations: None

Test Case Title	Skills Submission
Objective	Verify successful submission of skills
Preconditions	Personal details already submitted
Test Steps	1. Navigate to <code>skills.php</code> 2. Enter valid skills information - Skill 1: Programming - Skill 2: Communication 3. Submit the form
Expected Result	- Success message displayed - Skills information stored in the database
Postconditions	- User's skills visible in the application
Test Report	Number of test cases executed: 1 Number of test cases passed: 1 Number of test cases failed: 0 Defects found: None Recommendations: None

Test Case Title	School Information Submission
Objective	Verify successful submission of school information
Preconditions	Personal details and skills information already submitted
Test Steps	1. Navigate to <code>SchoolInformation.php</code> 2. Enter valid school information - Last School Attended: XYZ High School - Highest Grade Passed: 12 - Final Year: 2022 - Subjects and Marks for 7 subjects 3. Submit the form
Expected Result	- Success message displayed - School information stored in the database
Postconditions	- User's school information visible in the application
Test Report	Number of test cases executed: 1 Number of test cases passed: 1 Number of test cases failed: 0 Defects found: None Recommendations: None

Test Case Title	Higher Education Submission
Objective	Verify successful submission of higher education details
Preconditions	Personal details, skills information, and school information already submitted
Test Steps	1. Navigate to HigherEducation.php 2. Enter valid higher education information - College Name: ABC University - Field of Study: Computer Science - Active: Yes - Completed: No 3. Submit the form
Expected Result	- Success message displayed - Higher education details stored in the database
Postconditions	- User's higher education information visible in the application
Test Report	Number of test cases executed: 1 Number of test cases passed: 1 Number of test cases failed: 0 Defects found: None Recommendations: None

Test Case ID	BU_002	Test Case Description	Test the company Login Functionality in TalentForge		
Created By	sherisan	Reviewed By	Yodin	Version	2.1

QA Tester's Log	Review comments from sherisan incorprate in version 2.1
------------------------	---

Tester's Name	sherisan	Date Tested	December 1, 2023	Test Case (Pass/Fail/Not Executed)	Pass
----------------------	----------	--------------------	------------------	---	------

S #	Prerequisites:
1	Access to Chrome Browser
2	
3	
4	

S #	Test Data
1	Userid = richfiled@gamil.com
2	Pass = richfield
3	
4	

Test Scenario	Verify on entering valid company user username and password, the user can login
----------------------	---

Step #	Step Details	Expected Results		Actual Results	Pass / Fail / Not executed / Suspended
1	Navigate to TalentforgeHomepage.php	Site should open		As Expected	Pass
2	Navigate to LoginPage.php	Site should open		As Expected	Pass
4	Navigate to CompanyAccount	Userlogin.php should open		As Expected	Pass
5	Enter Username & Password	Credential can be entered		As Expected	Pass
6	Click Submit	user is logged in and taken to company account home page		As Expected	Pass