TRƯỜNG CAO ĐẲNG CÔNG NGHỆ THỦ ĐỨC

KHOA CÔNG NGHỆ THÔNG TIN



PRJ – REPORT

BÁO CÁO DEMO BẢO MẬT

**HƯỚNG DẪN CÁCH VÁ LỖI BẢO MẬT SQL INJECTION ID**

Giảng Viên: Phan Thanh Nhuần

Nhóm D:

1.Trần Phong Nhã - 18211TT1000

2.Nguyễn Phương Tân - 18211TT2680

3.Trần Triều Thuận - 18211TT0530

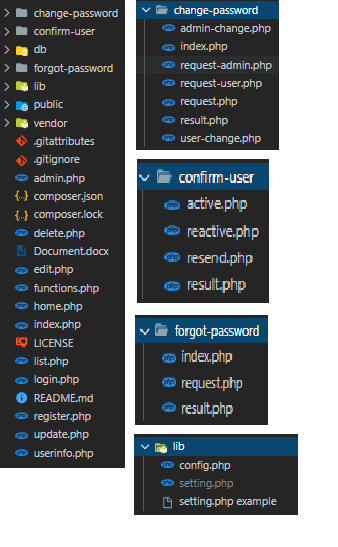
Ngày 26/10/2020

**Báo Cáo Thực Hành DEMO nhóm D**

Các chức năng được phân công như sau:

|  |  |
| --- | --- |
| Các chức năng | Người thực hiện |
| - Vá lỗi bảo mật (không thực hiện action theo id) trong Edit, Delete, View của User - Quên mật khẩu (xác nhận email) - Thay đổi mật khẩu (gửi thông báo qua email)  + Sort theo username | Nguyễn Phương Tân |
| - Kiểm tra user và email nếu trùng user hoặc email trong database thì báo lỗi khi đăng ký. + Đăng nhập bằng username hoặc email - Tìm kiếm theo email - Ràng buộc mật khẩu khi đăng ký phải có mức độ bảo mật cao. (> 8 ký tự, chữ hoa chữ thường, ký tự đặc biệt) | Trần Phong Nhã |
| - Tìm kiếm theo Fullname - Đăng ký tài khoản (Phải dùng xác nhận kích hoạt tài khoản qua mail) - Gửi lại xác nhận tài khoản (khi người dùng bỏ qua cảnh báo lần đầu lúc đăng ký) + Sort theo fullname | Trần Triều Thuận |

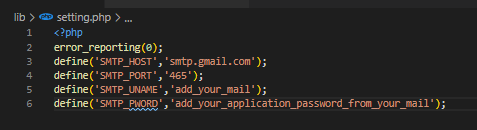
**Cấu trúc Source code**



**Một số hàm, file quan trọng được xây dựng phục vụ trong Source code này**

**lib/setting.php**

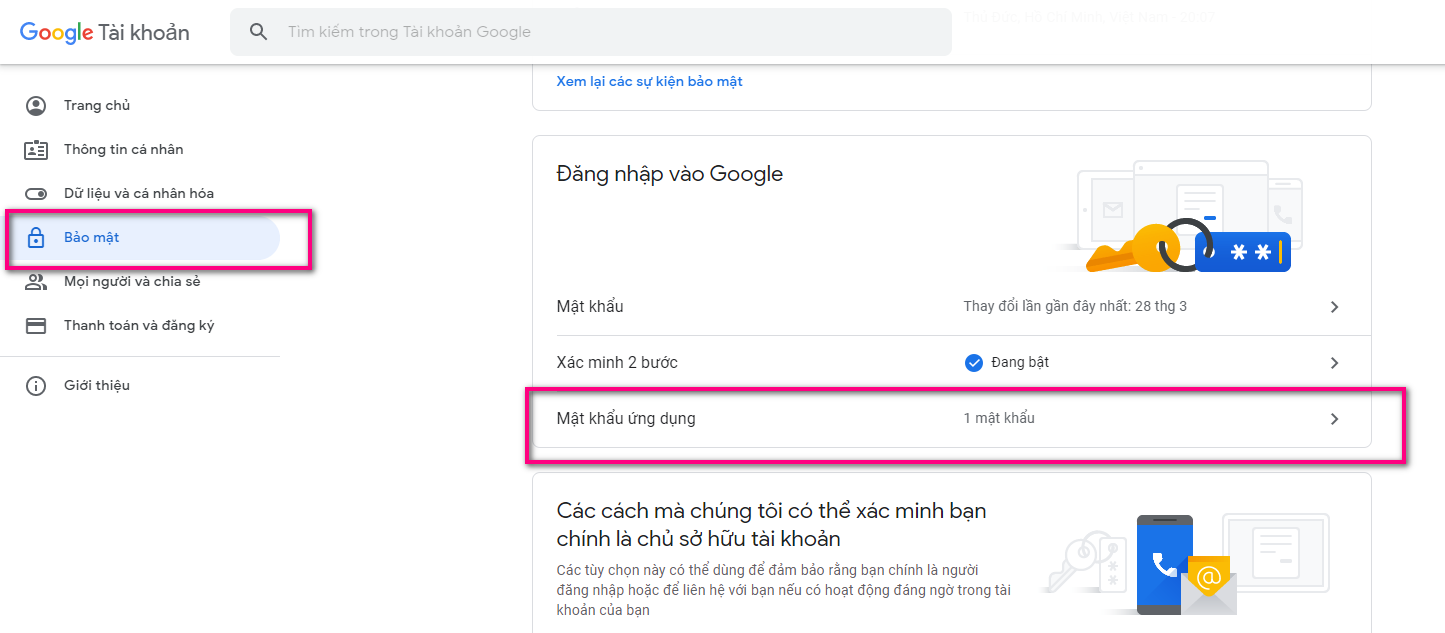
Mục đích file này sử dụng cho phần cấu hình gửi mail cho các chức năng. Người dùng phải tạo mật khẩu ứng dụng tại mục “**Bảo Mật**” của email và chỉnh sửa tại file này để chức năng gửi mail hoạt động.



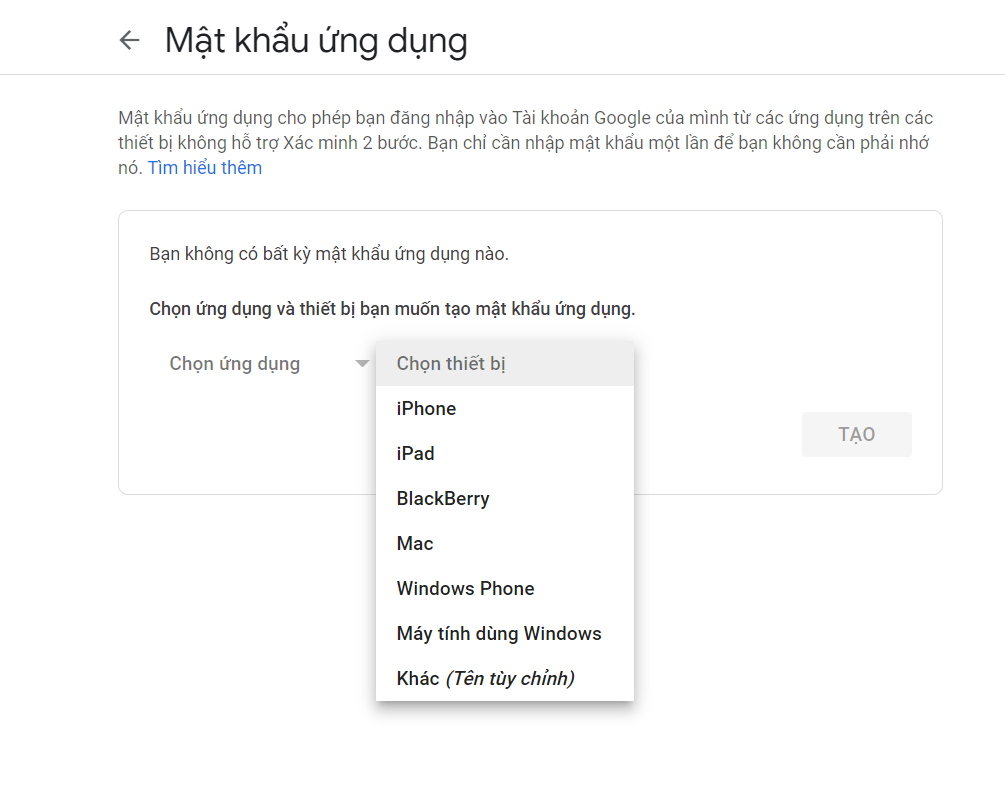
Từ source code, hãy tạo một file **setting.php** trong thư mục **lib**. (Đã có file **đuôi example** mẫu trong thu mục **lib**. hãy copy file đó và đổi tên lại)

**Cách lấy mật khẩu ứng dụng cho email**

Đăng nhập vào tài khoản email và đi đến phần bảo mật, chọn mục mật khẩu ứng dụng để tạo mới.



Chọn vào mục “Khác (tên tùy chỉnh)” đặt tên cho ứng dụng và sau đó lưu lại.



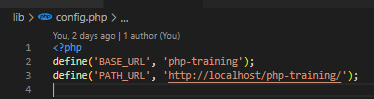
Tiếp theo, copy mật khẩu đó và dán vào thay thế lại hằng **SMTP\_PWORD**, còn hằng **SMPT\_UNAME** thì nhập đầy đủ email mà bạn vừa tạo mật khẩu.

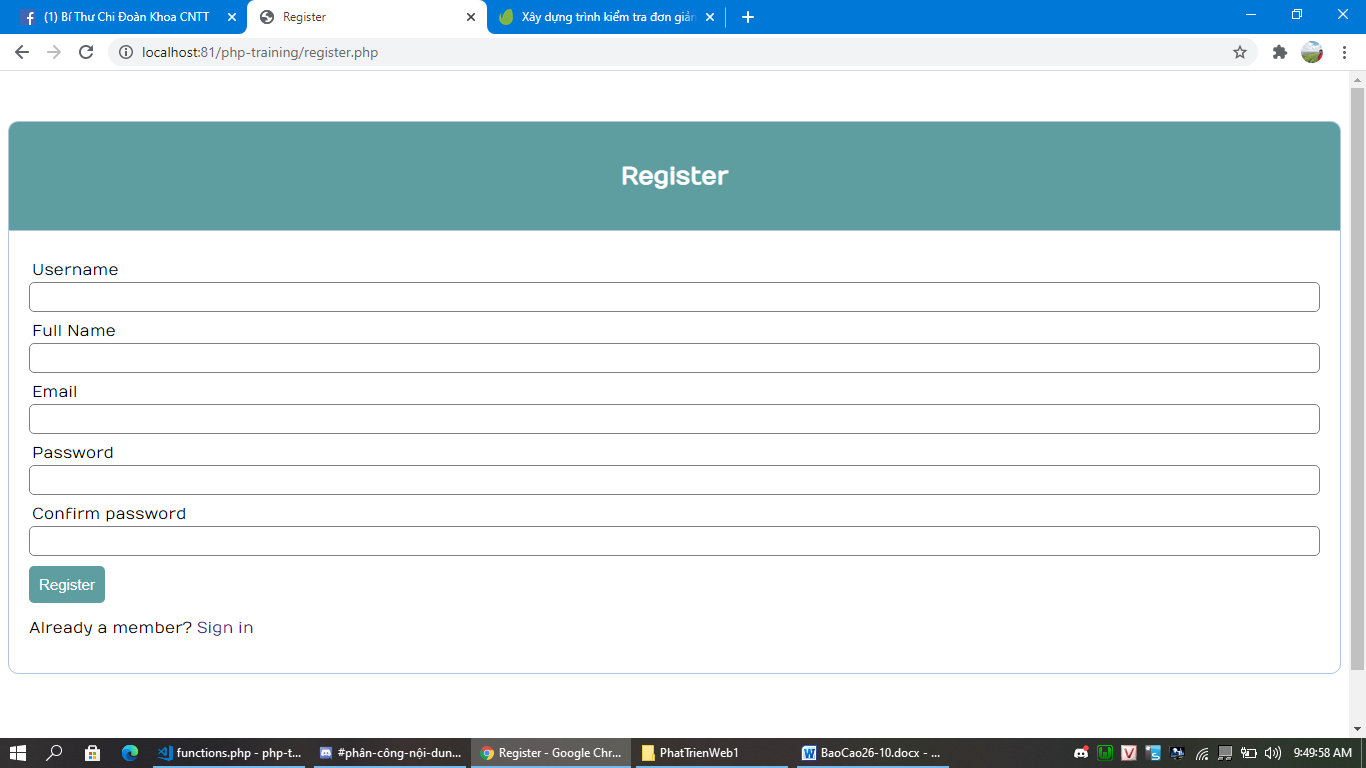
**Function.php**

File này chứa các hàm chức năng được tái sử dụng liên tục trong các tính năng của website.

**lib/config.php**

Tại file này người dùng source code phải chỉnh sửa lại **PATH URL** phù hợp với host server.

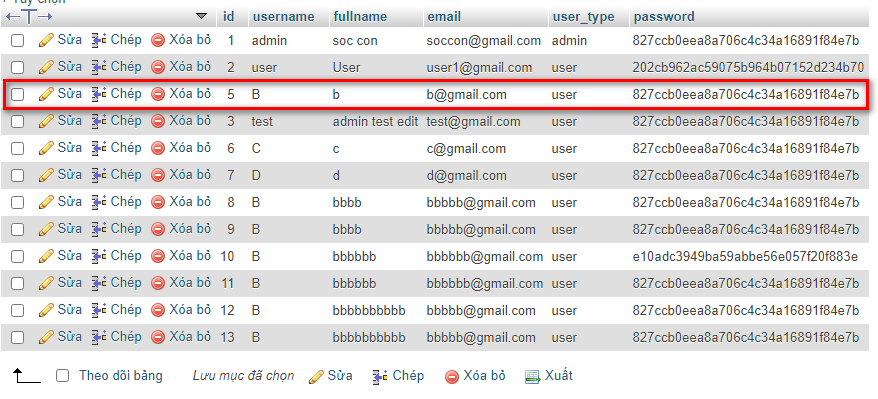


1. **Kiểm tra User và Email Khi đăng ký thành viên mới**

Giao diện register. cho phép người dùng đăng ký tài khoản.

**Trường hợp 1:** trùng username khi đăng ký

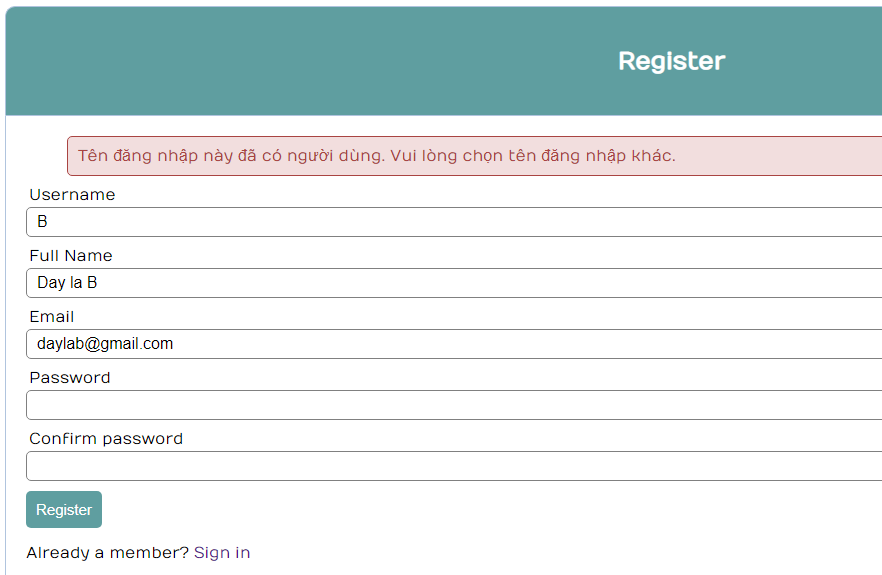
Khi người dùng đăng ký trùng với username có trong database thì sẽ báo lỗi.



Đây là ví dụ: ta có database có username là B đã đăng ký thành công. Nếu ta đăng ký thêm một tài khoản có username củng là B tại form register.



Nếu ta nhấn button register thì sẽ báo lỗi do trong database đã có username B.



Code thực hiện việc kiểm tra username:

if (isset($\_POST['register\_btn'])) {

    global $conn, $errors, $username, $fullname, $email;

    $username = escape($\_POST['username']);

    $email = escape($\_POST['email']);

    if (mysqli\_num\_rows(mysqli\_query($conn, "SELECT username FROM users WHERE username='$username'")) != 0) {

        array\_push($errors, "Username đã tồn tại");

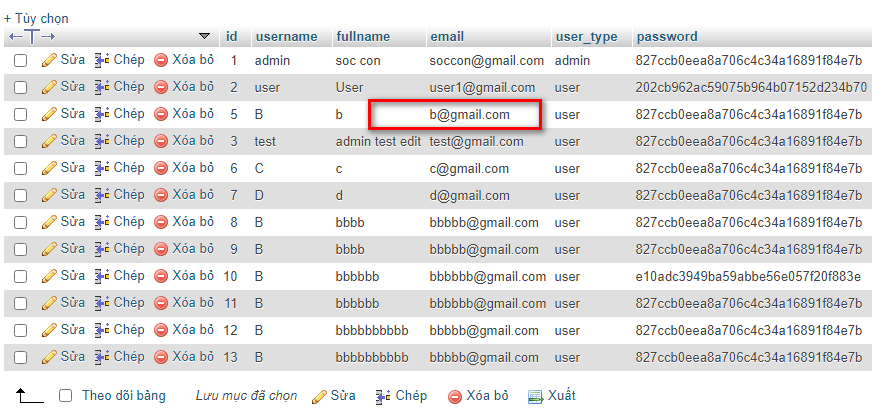
    } elseif (mysqli\_num\_rows(mysqli\_query($conn, "SELECT email FROM users WHERE email='$email'"))) {

        array\_push($errors, "Email đã tồn tại");

    } else register();

}

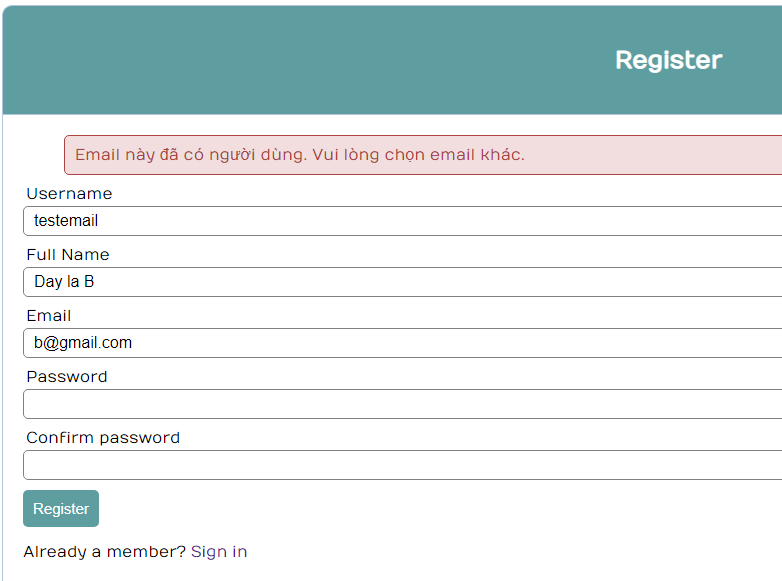
**Trường hợp 2:** củng giống như trường hợp 1. Nếu email người dùng đăng ký trùng với email đã tồn tại trong database thì sẽ báo lỗi cho người dùng khi đăng ký.



Trong database đã có email: [b@gmail.com](mailto:b@gmail.com) . nếu ta đăng ký thêm 1 email trùng với email trong database thì sẽ báo lỗi.



Khi nhấn button **Register.**



Code thực hiện việc kiểm tra email

if (isset($\_POST['register\_btn'])) {

    global $conn, $errors, $username, $fullname, $email;

    $username = escape($\_POST['username']);

    $email = escape($\_POST['email']);

    if (mysqli\_num\_rows(mysqli\_query($conn, "SELECT username FROM users WHERE username='$username'")) != 0) {

        array\_push($errors, "Username đã tồn tại");

    } elseif (mysqli\_num\_rows(mysqli\_query($conn, "SELECT email FROM users WHERE email='$email'"))) {

        array\_push($errors, "Email đã tồn tại");

    } else register();

}

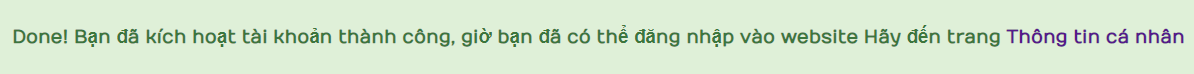
Phần kiểm tra username và email khi đăng ký này được thực hiện trong file **function.php.**

1. **Xác nhận kích hoạt tài khoản xác nhận email sau khi vừa đăng ký.**

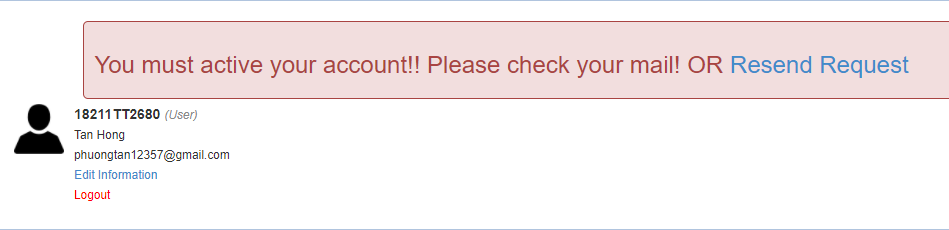
Sau khi người dùng đăng ký tài khoản xong sẽ có email thông báo yêu cầu xác nhận email cho người dùng.



Khi người dùng nhấn vào button **VERIFY ACCOUNT** tài khoản của người dùng sẽ được Kích hoạt và có thông báo tren trình duyệt như sau:



Nếu người dùng không vào email kiểm tra thư và kích hoạt thì lúc xem thông tin cá nhân sẽ thấy dòng cảnh báo nhưu bên dưới:

 Code thực hiện việc **VERIFY ACCOUNT**

if (isset($\_SESSION['activeCode']) && $\_SESSION['activeCode'] == $\_GET['code']) {

    $select\_user\_option = array(

        'order\_by' => 'id'

    );

    $user\_need\_activate = get\_by\_options('users', $select\_user\_option);

    foreach ($user\_need\_activate as $user) {

        if ($user['verificationCode'] == $\_GET['code']) {

            $verifi\_id\_user = $user['id'];

        }

    }

    if (!isset($verifi\_id\_user)) {

        exit;

    }

    $user\_edit = array(

        'id' => $verifi\_id\_user,

        'status' => 1

    );

    save('users', $user\_edit);

    $mess = "<strong>Done!</strong> Bạn đã kích hoạt tài khoản thành công, giờ bạn đã có thể đăng nhập vào website Hãy đến trang <a href='../index.php'>Thông tin cá nhân</a>";

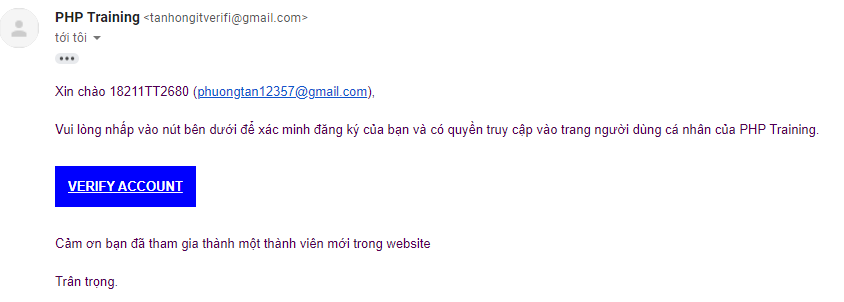
}

require('result.php');

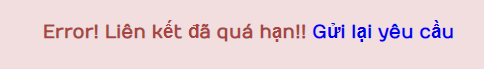
1. **Gửi lại xác nhận tài khoản (khi người dùng bỏ qua cảnh báo yêu cầu kích hoạt lần đầu sau khi đăng ký)**

Trường hợp nếu người dùng vừa đăng ký xong nhưng không vào hộp thư email để nhấn link kích hoạt mà người dùng đó đã tắt trình duyệt hiện hành. Khi này session trong trình duyệt sẽ bị hủy cho nên link kích hoạt trong email sẽ hết hạn.

Nếu người dùng lúc này mà nhấn vào link trong email thì sẽ có thông báo như sau:



Kết quả:



Code thực hiện kiểm tra Session để hiển thị ra thông báo liên kết quá hạn:

if (!isset($\_GET['code']) || empty($\_GET['code']) || !isLoggedIn()) {

    header('location: ../index.php');

}elseif(!isset($\_SESSION['activeCode'])){

    $mess\_er = "<strong>Error!</strong> Liên kết đã quá hạn!! <a href='/resend.php'>Gửi lại yêu cầu</a>";

}

if (isset($\_SESSION['activeCode']) && $\_SESSION['activeCode'] == $\_GET['code']) {

    $select\_user\_option = array(

        'order\_by' => 'id'

    );

    $user\_need\_activate = get\_by\_options('users', $select\_user\_option);

    foreach ($user\_need\_activate as $user) {

        if ($user['verificationCode'] == $\_GET['code']) {

            $verifi\_id\_user = $user['id'];

        }

    }

    if (!isset($verifi\_id\_user)) {

        exit;

    }

    $user\_edit = array(

        'id' => $verifi\_id\_user,

        'status' => 1

    );

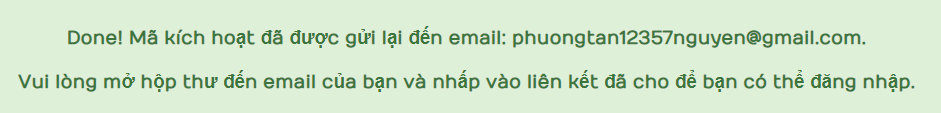
    save('users', $user\_edit);

    $mess = "<strong>Done!</strong> Bạn đã kích hoạt tài khoản thành công, giờ bạn đã có thể đăng nhập vào website Hãy đến trang <a href='../index.php'>Thông tin cá nhân</a>";

}

require('result.php');

Khi này người dùng dằng nhập vào tài khoản và nhấn vào text link “**Gửi lại yêu cầu**” thì sẽ có dòng thông báo như sau:



Người dùng tiếp tục vào lại email để nhấn link kích hoạt thì khi này sẽ kích hoạt thành công.

Code thực hiện việc **REACTIVE ACCOUNT**

if (isset($\_SESSION['resend\_confirm\_user']) && $\_SESSION['resend\_confirm\_user'] == $\_GET['code']) {

    $select\_user\_option = array(

        'order\_by' => 'id'

    );

    $user\_need\_activate = get\_by\_options('users', $select\_user\_option);

    foreach ($user\_need\_activate as $user) {

        if ($user['verificationCode'] == $\_GET['code']) {

            $verifi\_id\_user = $user['id'];

        }

    }

    if (!isset($verifi\_id\_user)) {

        exit;

    }

    $user\_edit = array(

        'id' => $verifi\_id\_user,

        'status' => 1

    );

    save('users', $user\_edit);

    $mess = "<strong>Done!</strong> Bạn đã kích hoạt tài khoản thành công, giờ bạn đã có thể đăng nhập vào website Hãy đến trang <a href='../index.php'>Thông tin cá nhân</a>";

}

require('result.php');

1. **Tìm kiếm theo Username, Email và Full name**

Ngoài việc search username thì chúng ta nên bổ sung thêm search bằng email va full name.

Câu truy vấn sql dùng để tìm kiếm kết hợp tìm kiếm giữa username, email và full name:

$query = "SELECT \* FROM users WHERE username LIKE '%$search%' OR fullname LIKE '%$search%' OR email LIKE '%$search%'";

Trong câu truy vấn query, ta thêm phần **OR fullname** và **OR email** vào nhằm kết hợp tìm kiếm theo nhiều cột khác nhau.

Code đầy đủ về chức năng search.

// Gán hàm addslashes để chống sql injection

isset($\_GET['search']) ? $search = addslashes($\_GET['search']) : $search = "";

$options\_search = array(

    'where' => "username LIKE '%" . ($search) . "%' or fullname like '%" . ($search) . "%'",

    'limit' => $limit,

    'offset' => $offset,

    'order\_by' => 'id ASC'

);

$query = "SELECT \* FROM users WHERE username LIKE '%$search%' OR fullname LIKE '%$search%' OR email LIKE  '%$search%'";

global $conn;

$sql = mysqli\_query($conn, $query);

$num = mysqli\_num\_rows($sql);

Tiếp tục để trả về giao diện người dùng chúng ta có đoạn lệnh:

<?php

if ($search == "") {

    header("location: list.php");

} elseif ($num > 0) {

    echo "$num kết quả trả về với từ khóa '<b>$search</b>'";

    while ($result = mysqli\_fetch\_assoc($sql)) { ?>

        <tr scope="row">

            <td><?php echo $result['id']; ?></td>

            <td><?php echo $result['username']; ?></td>

            <td><?php echo $result['fullname']; ?></td>

            <td><?php echo $result['email']; ?></td>

            <td>

                <a href="userinfo.php?user\_id=<?= getLink($result['id']) ?>"><i class="fa fa-eye" aria-hidden="true"></i></a>

                <a href="edit.php?edit=<?= getLink($result['id']) ?>"><i class="fa fa-pencil-square-o" aria-hidden="true"></i></a>

                <a onclick="return confirm('Are you sure to delete?')" href="delete.php?user\_id=<?= getLink($result['id']) ?>"><i class="fa fa-times" aria-hidden="true"></i></a>

            </td>

        </tr>

<?php  }

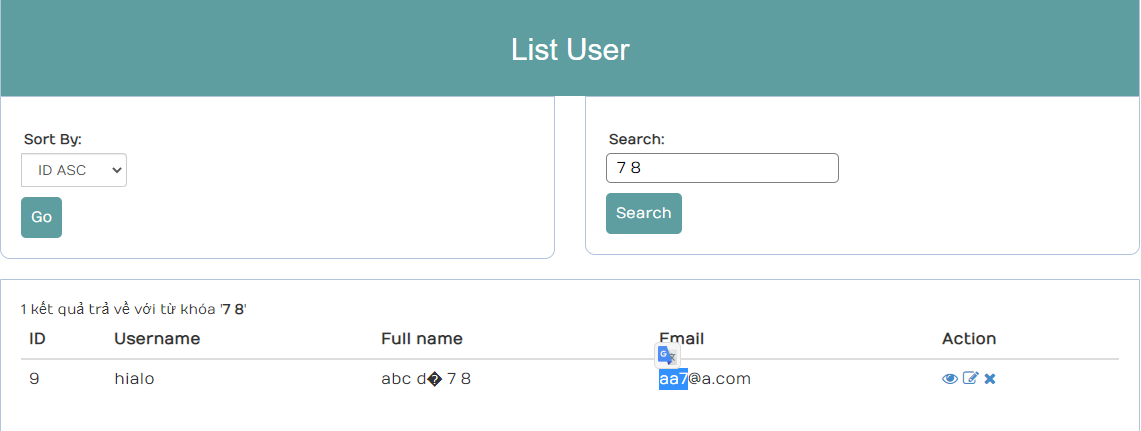
} else {

    echo "Khong tim thay ket qua!";

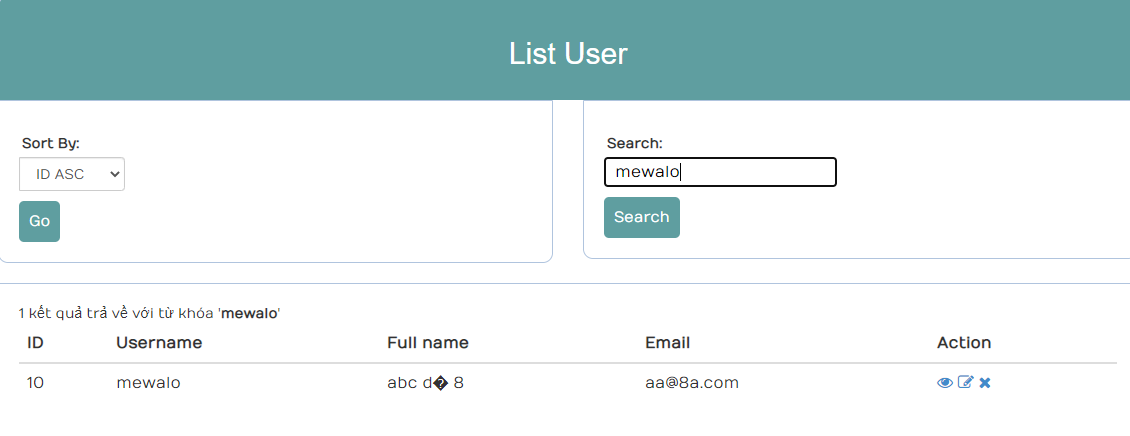
}

Ví dụ tại giao diện người dùng:

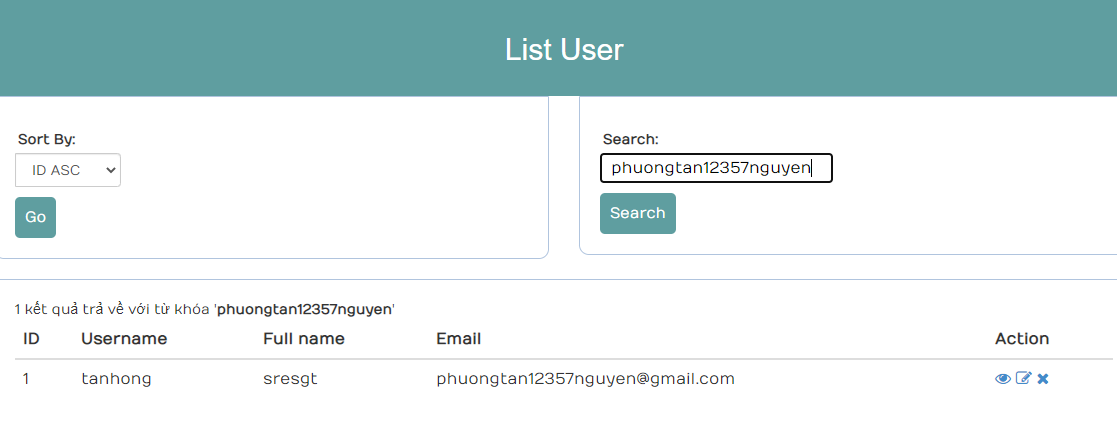
1. Tìm kiếm theo full name



1. Tìm kiếm tho username



1. Tìm kiếm theo email



1. **Cách vá lỗi bảo mật sql injection khi thao tác các hành động với ID**

Nếu sử dụng trực tiếp **ID** để thực hiện một số hành động như thêm, xóa, xem chi tiết người dùng thì sẽ dính vào lỗi bảo mật nghiêm trọng **sql injection**.

Cho nên chúng ta sẽ sử dụng phương pháp tạo một chuỗi **md5** cho **ID** đó rồi lưu vào biến **session** và liên tục random giá trị mới của nó mỗi khi truy cập lại trang.

**Phương pháp làm như sau**:

Thứ nhất, xây dựng một hàm **getLink()** để liên tục thay đổi **giá trị** **ID** thành một chuỗi dài ký tự. Hàm này sẽ nằm trong file **function.php** để có thể tái sử dụng nhiều lần.

//encode id

function getLink($id)

{

    $random = md5(uniqid($id));

    $\_SESSION['links\_edit'][$random] = $\_SESSION['info\_user\_id'][$random] = $id;

    return "$random";

}

Chúng ta thấy trong hàm có sử dụng hai hàm **md5()** kết hợp với **uniqid()** nhằm tạo ra mã chuỗi ngẫu nhiên duy nhất và thay liên tục sau mỗi lần load trang hoặc khác nhau theo từng **ID** của từng user.

Tài liệu tham khảo về uniqid:

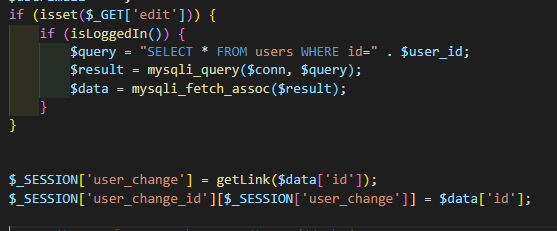
* <https://www.php.net/manual/en/function.uniqid.php>
* <https://www.w3schools.com/php/func_string_md5.asp>

Giá trị **ID** sẽ được lưu vào Session theo từng **key** là chuỗi vừa được tạo duy nhất theo mã **md5()** **ID**.

Nói một cách đơn giản hơn rằng chúng ta sẽ liên tục tạo ra nhưng chuỗi mới khác nhau nhằm thay thế cho từng **ID**, và mỗi khi load lại trang web thì chuỗi thay thế này sẽ liên tục bị thay đổi giúp cho hacker khó phát hiện ra **ID** thật sự là gì.

**Cách áp dụng hàm md5 và uniqid vào thực tiễn như sau**:

Đầu tiên, ta sẽ dùng kỹ thuật truy vấn để lấy ra mảng **$data** chứa thông tin của người dùng. Sau đó ta sử dụng hàm **getLink()** để chuyển hóa **$data[‘id’]** (là **ID** của User) thành dạng chuỗi. (Sẽ bị thay đổi liên tục mỗi khi load lại trang) và lưu nó vào một biến **session** có tên là **user\_change.**

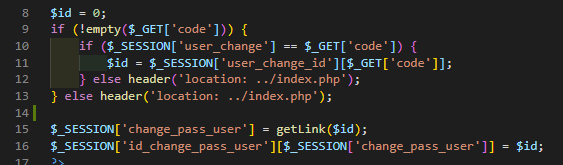


Chúng ta sẽ lưu thêm một session mới tên **user\_change\_id** có Key là biến **$\_SESSION['user\_change']** và giá trị là **$data[‘id’]**

Tiếp theo ta sẽ truyền nó vào đường dẫn để đi đến hành động mà ta muốn. (Ở ví dụ này sẽ dùng hành động thay đổi mật khẩu cho mỗi user).



Tại file **user-change.php** ta sẽ có một đoạn lệnh kiểm tra như sau:



Chúng ta sẽ kiểm tra biến **session** vừa lưu và chuỗi code trên đường dẫn có bằng hay không. Nếu bằng ta sẽ gọi ra được **ID** (là **value** theo **key** là chuỗi code đó.)

Một lần nữa, chúng ta sẽ tạo một chuỗi mới cho **ID** lưu vào biến session có tên là **change\_pass\_user** nhằm đưa chuỗi nào vào form nhập liệu bên dưới giúp dánh một số người dùng sử dụng f12 thay đổi giá trị action trong form để tấn công vào lỗ hổng bảo mật trong chương trình.

<form action="request-user.php" method="post" style="text-align: center;">

            <input type="hidden" name="change" value="<?= $\_SESSION['change\_pass\_user'] ?>">

            <div class="input-group">

                <label>Enter your current password!</label><br>

                <input required type="password" name="currentPassword" placeholder="Enter Password" autofocus />

            </div><br>

            <div class="input-group">

                <label>Enter your new password!</label><br>

                <input required type="password" name="newPassword" placeholder="Enter Password" autofocus />

            </div><br>

            <div class="input-group">

                <label>Please confirm your new password!</label><br>

                <input required type="password" name="confirmNewPassword" placeholder="Enter Password Again" autofocus />

            </div>

            <br>

            <button type="button" class="btn btn-info" onClick="javascript:history.go(-1)">Back</button>

            <button type="submit" class="btn btn-info">Send Request</button>

            <br>

        </form>

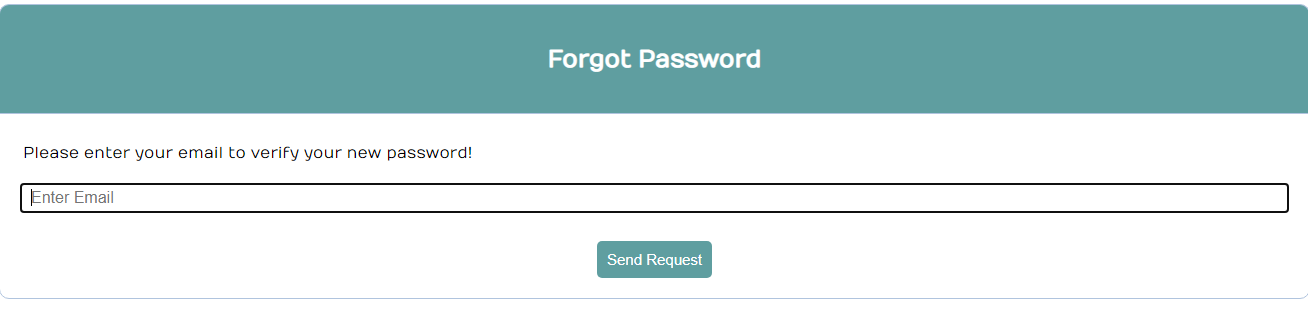
Ở form trên cúng ta sẽ gọi biến **session** vừa tạo tại đoạn lệnh **<input type="hidden" name="change" value="<?= $\_SESSION['change\_pass\_user'] ?>">**

1. **Chức năng quên mật khẩu (Forgot password)**

Khi người dùng quên mật khẩu tài khoản đăng nhập của họ, người dùng nhấn vào “**Forgot password?”**



Sau khi nhấn sẽ chuyển đến form nhập email của họ.



Code form forgot password.

<html>

<head>

<title>Forgot Password</title>

<link rel="stylesheet" href="../public/css/styles.css">

</head>

<body>

<div class="container">

<!-- notification message -->

<?php if (isset($\_SESSION['forgot\_pass'])) : ?>

<div class="error danger" style="text-align: center;">

<h3>

<?php

echo $\_SESSION['forgot\_pass'];

unset($\_SESSION['forgot\_pass']);

?>

</h3>

</div>

<?php endif ?>

<?php if (isset($\_SESSION['forgot\_pass\_suc'])) : ?>

<div class="error success" style="text-align: center;">

<h3>

<?php

echo $\_SESSION['forgot\_pass\_suc'];

unset($\_SESSION['forgot\_pass\_suc']);

?>

</h3>

</div>

<?php endif ?>

<div class="header">

<h2>Forgot Password</h2>

</div>

<form action="request.php" method="post" style="text-align: center;">

<div class="input-group">

<label>Please enter your email to verify your new password!</label><br>

<input required type="email" name="email" placeholder="Enter Email" autofocus />

</div>

<br>

<button type="submit" class="btn btn-info">Send Request</button>

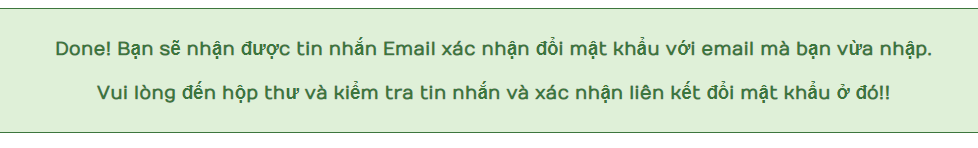
</form>

</div>

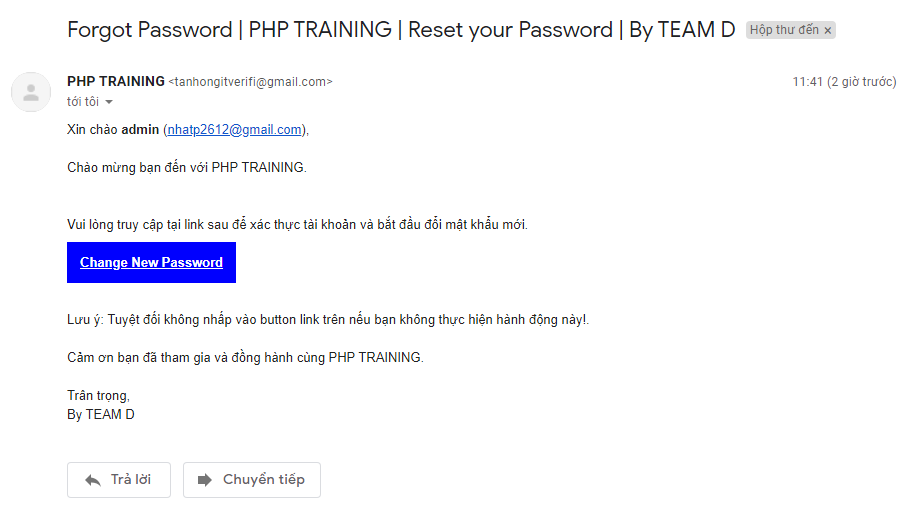
</body>

</html>

Sau khi nhấn kích hoạt với gmail đã có tồn tại trong dtabase thì sẽ nhận được thông báo như sau:



Ở form này yêu cầu người dùng phải nhập email để gửi đường dẫn xác thực kích hoạt tài khoản lên Email và người dùng phải nhấn vào button kích hoạt trong mail.



Code cấu hình phần mục gửi mail:

//send mail

    include 'lib/config.php';

    require 'vendor/autoload.php';

    include 'lib/setting.php';

    $mail = new PHPMailer(true);

    try {

        $verificationCode\_iduser = md5(uniqid("Email của bạn chưa active. Nhấn vào đây để active nhé!", true));

        $verificationCode = PATH\_URL . "confirm-user/active.php?code=" . $verificationCode\_iduser;

        //content

        $\_SESSION['activeCode'] = $verificationCode\_iduser;

        $\_SESSION['verificationLink'] = $verificationCode;

        $htmlStr = "";

        $htmlStr .= "Xin chào " . $username . ' (' . $email . "),<br /><br />";

        $htmlStr .= "Vui lòng nhấp vào nút bên dưới để xác minh đăng ký của bạn và có quyền truy cập vào trang người dùng cá nhân của PHP Training.<br /><br /><br />";

        $htmlStr .= "<a href='{$\_SESSION['verificationLink']}' target='\_blank' style='padding:1em; font-weight:bold; background-color:blue; color:#fff;'>VERIFY ACCOUNT</a><br /><br /><br />";

        $htmlStr .= "Cảm ơn bạn đã tham gia thành một thành viên mới trong website<br><br>";

        $htmlStr .= "Trân trọng.<br />";

        //Server settings

        $mail->CharSet = "UTF-8";

        $mail->SMTPDebug = 0; // Enable verbose debug output (0 : ko hiện debug, 1 hiện)

        $mail->isSMTP(); // Set mailer to use SMTP

        $mail->Host = SMTP\_HOST; // Specify main and backup SMTP servers

        $mail->SMTPAuth = true; // Enable SMTP authentication

        $mail->Username = SMTP\_UNAME; // SMTP username

        $mail->Password = SMTP\_PWORD; // SMTP password

        $mail->SMTPSecure = 'ssl'; // Enable TLS encryption, `ssl` also accepted

        $mail->Port = SMTP\_PORT; // TCP port to connect to

        //Recipients

        $mail->setFrom(SMTP\_UNAME, "PHP Training");

        $mail->addAddress($\_POST['email'], $email); // Add a recipient | name is option tên người nhận

        $mail->addReplyTo(SMTP\_UNAME, 'PHP TRAINING');

        $mail->isHTML(true); // Set email format to HTML

        $mail->Subject = 'Verification Users | PHP Training';

        $mail->Body = $htmlStr;

        $mail->AltBody = $htmlStr; //None HTML

        $result = $mail->send();

        if (!$result) {

            $error = "Có lỗi xảy ra trong quá trình gửi mail";

        }

    } catch (Exception $e) {

        echo 'Message could not be sent. Mailer Error: ', $mail->ErrorInfo;

    }

    $verificationCode\_add = array(

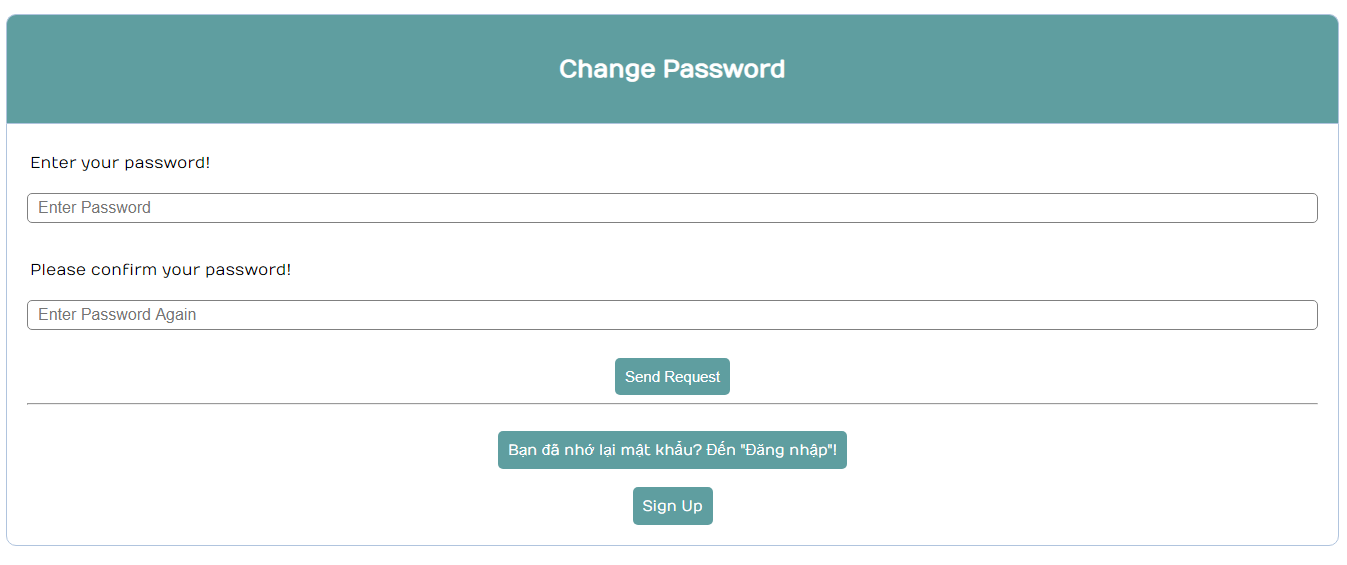
        'id' => $user\_id,

        'verificationCode' => $verificationCode\_iduser

    );

    save('users', $verificationCode\_add);

Sau khi người dùng truy cập mail và click vào button “**Change new password**”. sẽ chuyển đến form **change password**



Code form change password

<?php

session\_start();

include('../functions.php');

$id = 0;

if (!empty($\_GET['code'])) {

if (isset($\_SESSION['forgot\_pass\_Code']) && $\_SESSION['forgot\_pass\_Code'] == $\_GET['code']) {

$id = $\_SESSION['forgot\_pass\_id'][$\_GET['code']];

}else header('location: ../index.php');

}else header('location: ../index.php');

$\_SESSION['change\_pass'] = getLink($id);

$\_SESSION['id\_change\_pass'][$\_SESSION['change\_pass']] = $id;

?>

<html>

<head>

<title>Change Password</title>

<link rel="stylesheet" href="../public/css/styles.css">

</head>

<body>

<div class="container">

<div class="header">

<h2>Change Password</h2>

</div>

<form action="request.php" method="post" style="text-align: center;">

<input type="hidden" name="change" value="<?= $\_SESSION['change\_pass'] ?>">

<div class="input-group">

<label>Enter your password!</label><br>

<input required type="password" name="newPassword" placeholder="Enter Password" autofocus />

</div><br>

<div class="input-group">

<label>Please confirm your password!</label><br>

<input required type="password" name="confirmNewPassword" placeholder="Enter Password Again" autofocus />

</div>

<br>

<button type="submit" class="btn btn-info">Send Request</button>

<br>

<?php if (!isLoggedIn()) : ?>

<hr><br>

<div class="input-group">

<a class="btn link" href="../login.php">Bạn đã nhớ lại mật khẩu? Đến "Đăng nhập"!</a>

</div>

<br>

<div class="input-group">

<a class="btn link" href="../register.php">Sign Up</a>

</div>

<?php endif; ?>

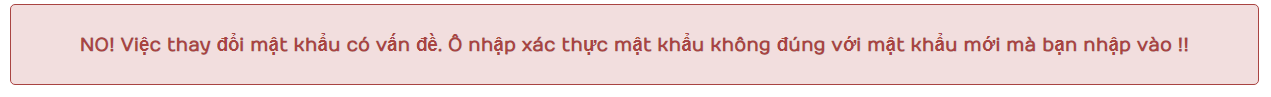
</form>

</div>

</body>

</html>

Ở form này cho phép người dùng nhập password mới và kiểm tra nếu người dùng confirm password mà không trùng với password mới thì sẽ báo lỗi và yêu cầu người dùng nhập lại.



if (!empty($\_POST['change'])) {

$id = $\_SESSION['id\_change\_pass'][$\_POST['change']];

$newpassword = md5($\_POST['newPassword']);

$confirmNewPassword = md5($\_POST['confirmNewPassword']);

$user = get\_a\_record('users', $id);

$email = $user['email'];

global $conn;

if ($newpassword == $user['password']) {

$mess = "<strong>NO!</strong> Việc thay đổi mật khẩu có vấn đề. Mật khẩu mới của bạn vừa nhập là mật khẩu của bạn hiện tại đó.";

} elseif (strlen($\_POST['newPassword']) < 8) {

$mess = "<strong>NO!</strong> Việc thay đổi mật khẩu thất bại. Mật khẩu bạn nhập phải dài từ 8 ký tự trở lên !!";

} elseif ($newpassword == $confirmNewPassword) {

$options = array(

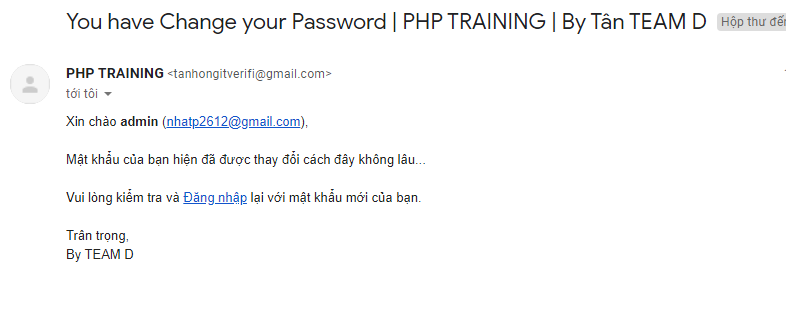
'id' => $id,

'password' => $newpassword

);

save('users', $options);

Nếu người dùng nhập đúng sẽ hiện thông báo thành công và sẽ gửi email thông báo đồng thời lưu mật khẩu đến dữ liệu người dùng đó trên database:



Code cấu hình email gửi báo đã thay đổi mật khẩu:

//send mail

require '../vendor/autoload.php';

include '../lib/setting.php';

$mail = new PHPMailer(true);

try {

//content

$htmlStr = "";

$htmlStr .= "Xin chào <strong>" . $user['username'] . '</strong> (' . $email . "),<br /><br />";

$htmlStr .= "Mật khẩu của bạn hiện đã được thay đổi cách đây không lâu...<br /><br />";

$htmlStr .= "Vui lòng kiểm tra và <a href='" . PATH\_URL . "login.php'>Đăng nhập</a></div> lại với mật khẩu mới của bạn.<br><br>";

$htmlStr .= "Trân trọng,<br />";

$htmlStr .= "By TEAM D<br />";

//Server settings

$mail->CharSet = "UTF-8";

$mail->SMTPDebug = 0; // Enable verbose debug output (0 : ko hiện debug, 1 hiện)

$mail->isSMTP(); // Set mailer to use SMTP

$mail->Host = SMTP\_HOST; // Specify main and backup SMTP servers

$mail->SMTPAuth = true; // Enable SMTP authentication

$mail->Username = SMTP\_UNAME; // SMTP username

$mail->Password = SMTP\_PWORD; // SMTP password

$mail->SMTPSecure = 'ssl'; // Enable TLS encryption, `ssl` also accepted

$mail->Port = SMTP\_PORT; // TCP port to connect to

//Recipients

$mail->setFrom(SMTP\_UNAME, "PHP TRAINING");

$mail->addAddress($email, $email); // Add a recipient | name is option tên người nhận

$mail->addReplyTo(SMTP\_UNAME, 'PHP TRAINING');

//$mail->addCC('CCemail@gmail.com');

//$mail->addBCC('BCCemail@gmail.com');

$mail->isHTML(true); // Set email format to HTML

$mail->Subject = 'You have Change your Password | PHP TRAINING | By Tân TEAM D';

$mail->Body = $htmlStr;

$mail->AltBody = $htmlStr; //None HTML

$result = $mail->send();

if (!$result) {

$error = "Có lỗi xảy ra trong quá trình gửi mail";

}

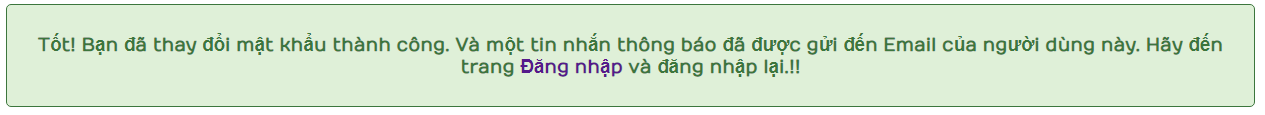
} catch (Exception $e) {

echo 'Message could not be sent. Mailer Error: ', $mail->ErrorInfo;

}

$mess\_success = '<strong>Tốt!</strong> Bạn đã thay đổi mật khẩu thành công. Và một tin nhắn thông báo đã được gửi đến Email của người dùng này. Hãy đến trang <a href="../login.php">Đăng nhập</a> và đăng nhập lại.!!';

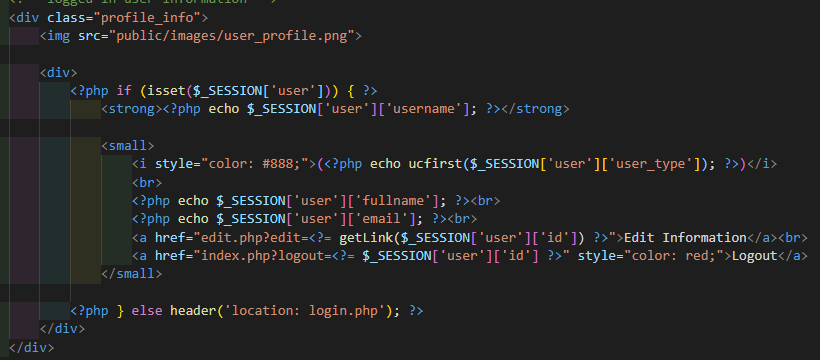
Trong thông báo có 1 đường dẫn đế trang login: “Đăng nhập” nếu người dùng click vào thì sẽ chuyển về trang login.

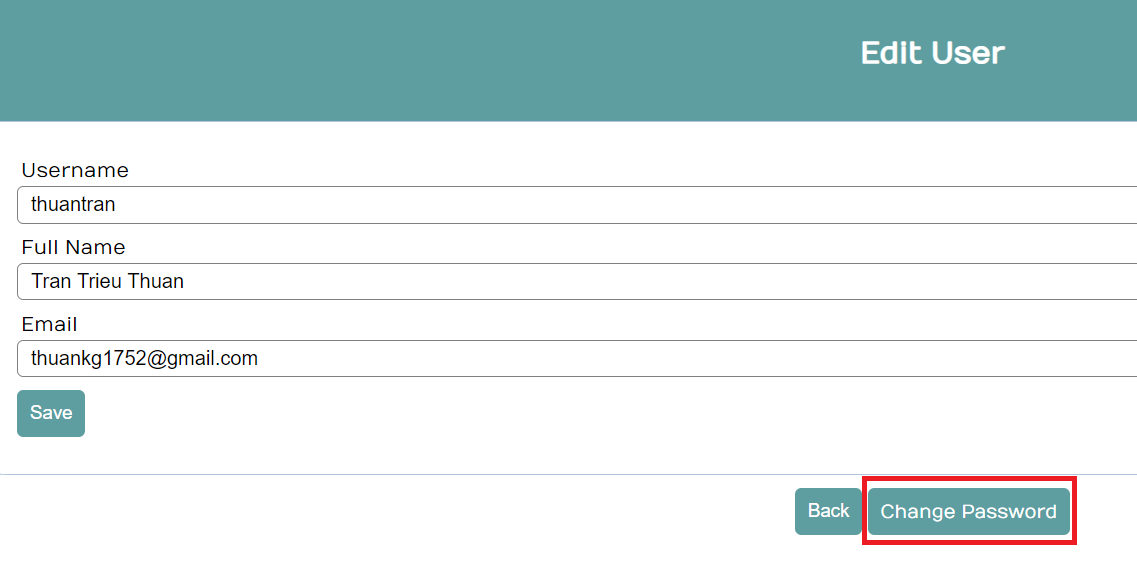


1. **Chức năng đổi mật khẩu với người dùng là User (Change password)**

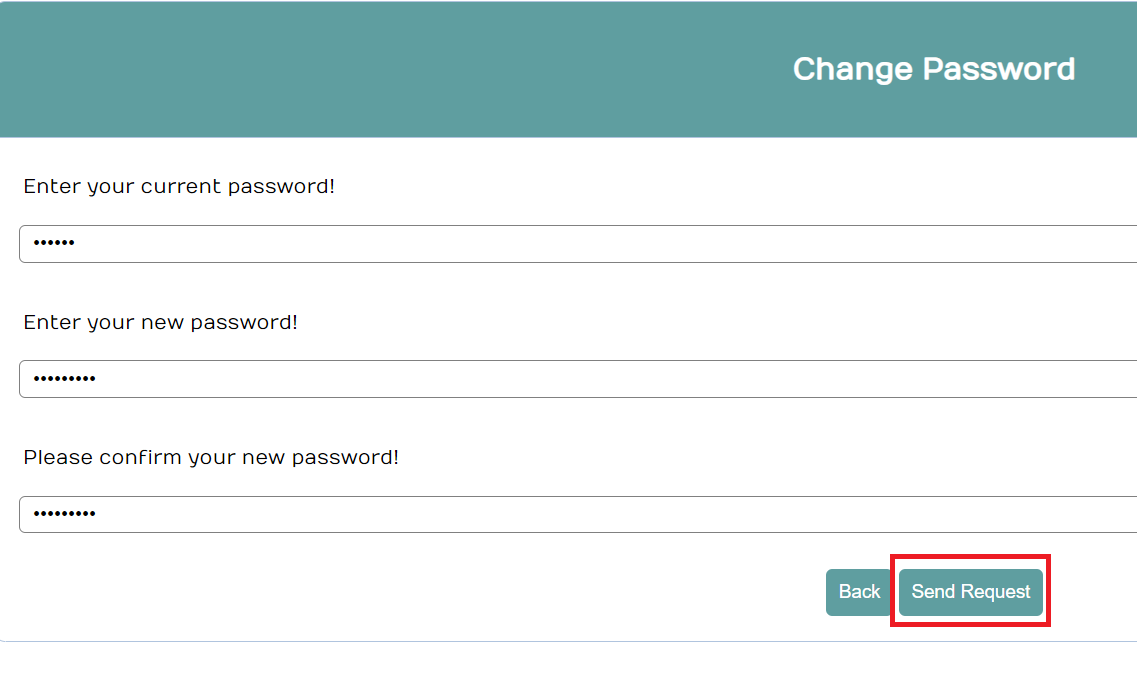
Sau khi người dùng đăng nhập thành công nếu muốn thay đổi thông tin cá nhân cần Click vào nút **Edit Information** sẽ được chuyển hướng đến trang **Edit User.**

Phần code dùng cho text link “**Edit Information**”.



Sau khi nhấn vào text link “**Edit Information**” thì sẽ chuyển hướng qua trang chỉnh sửa thông tin chi tiết người dùng.

Lúc này người dùng muốn thay đổi mật khẩu có thể click vào nút **Change password** sau khi click người dùng sẽ được chuyển đến giao diện đổi mật khẩu. Người dùng sẽ phải nhập mật khẩu cũ, mật khẩu mới, nhập lại mật khẩu mới muốn đổi.



Phần code dùng để kiểm tra password nhập vào:

if (!empty($\_POST['change'])) {

    $id = $\_SESSION['id\_change\_pass\_user'][$\_POST['change']];

    $currentPassword = md5($\_POST['currentPassword']);

    $newpassword = md5($\_POST['newPassword']);

    $confirmNewPassword = md5($\_POST['confirmNewPassword']);

    $user = get\_a\_record('users', $id);

    $email = $user['email'];

    global $conn;

    if ($user['password'] != $currentPassword) {

        $mess = "<strong>NO!</strong> Việc thay đổi mật khẩu có vấn đề. Mật khẩu hiện tại không đúng.<br><button type='button' class='btn btn-info' onClick='javascript:history.go(-1)'>Back</button> ";

    } elseif ($newpassword == $user['password']) {

        $mess = "<strong>NO!</strong> Việc thay đổi mật khẩu có vấn đề. Mật khẩu mới của bạn vừa nhập là mật khẩu của bạn hiện tại đó.<br><button type='button' class='btn btn-info' onClick='javascript:history.go(-1)'>Back</button> ";

    } elseif (strlen($\_POST['newPassword']) < 8) {

        $mess = "<strong>NO!</strong> Việc thay đổi mật khẩu thất bại. Mật khẩu bạn nhập phải dài từ 8 ký tự trở lên !!<br><button type='button' class='btn btn-info' onClick='javascript:history.go(-1)'>Back</button> ";

    } elseif ($newpassword == $confirmNewPassword) {

        $options = array(

            'id' => $id,

            'password' => $newpassword

        );

        save('users', $options);

        //send mail

        require '../vendor/autoload.php';

        include '../lib/setting.php';

        $mail = new PHPMailer(true);

        try {

            //content

            $htmlStr = "";

            $htmlStr .= "Xin chào <strong>" . $user['username'] . '</strong> (' . $email . "),<br /><br />";

            $htmlStr .= "Mật khẩu của bạn hiện đã được thay đổi cách đây không lâu...<br /><br />";

            $htmlStr .= "Vui lòng kiểm tra và <a href='" . PATH\_URL . "login.php'>Đăng nhập</a></div> lại với mật khẩu mới của bạn.<br><br>";

            $htmlStr .= "Trân trọng,<br />";

            $htmlStr .= "By TEAM D<br />";

            //Server settings

            $mail->CharSet = "UTF-8";

            $mail->SMTPDebug = 0; // Enable verbose debug output (0 : ko hiện debug, 1 hiện)

            $mail->isSMTP(); // Set mailer to use SMTP

            $mail->Host = SMTP\_HOST;  // Specify main and backup SMTP servers

            $mail->SMTPAuth = true; // Enable SMTP authentication

            $mail->Username = SMTP\_UNAME; // SMTP username

            $mail->Password = SMTP\_PWORD; // SMTP password

            $mail->SMTPSecure = 'ssl'; // Enable TLS encryption, `ssl` also accepted

            $mail->Port = SMTP\_PORT; // TCP port to connect to

            //Recipients

            $mail->setFrom(SMTP\_UNAME, "PHP TRAINING");

            $mail->addAddress($email, $email);     // Add a recipient | name is option tên người nhận

            $mail->addReplyTo(SMTP\_UNAME, 'PHP TRAINING');

            //$mail->addCC('CCemail@gmail.com');

            //$mail->addBCC('BCCemail@gmail.com');

            $mail->isHTML(true); // Set email format to HTML

            $mail->Subject = 'You have Change your Password | PHP TRAINING | By Tân TEAM D';

            $mail->Body = $htmlStr;

            $mail->AltBody = $htmlStr; //None HTML

            $result = $mail->send();

            if (!$result) {

                $error = "Có lỗi xảy ra trong quá trình gửi mail";

            }

        } catch (Exception $e) {

            echo 'Message could not be sent. Mailer Error: ', $mail->ErrorInfo;

        }

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

            unset($\_SESSION['user']);

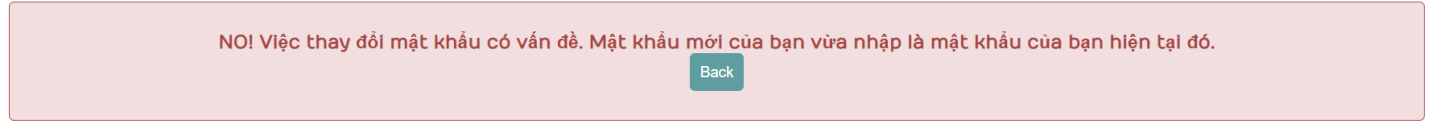
        }

        $mess\_success = '<strong>Tốt!</strong> Bạn đã thay đổi mật khẩu thành công. Và một tin nhắn thông báo đã được gửi đến Email của người dùng này. Hãy đến trang <a href="../login.php">Đăng nhập</a> và đăng nhập lại.!!';

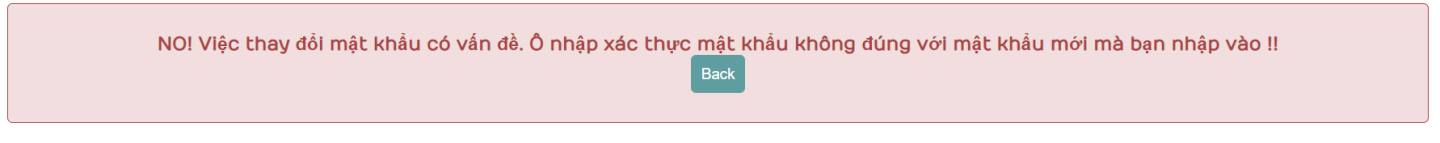
    } else $mess = '<strong>NO!</strong> Việc thay đổi mật khẩu có vấn đề. Ô nhập xác thực mật khẩu không đúng với mật khẩu mới mà bạn nhập vào !!<br><button type="button" class="btn btn-info" onClick="javascript:history.go(-1)">Back</button> ';

} else header('location: index.php');

Khi người dùng nhập trùng với mật khẩu cũ:



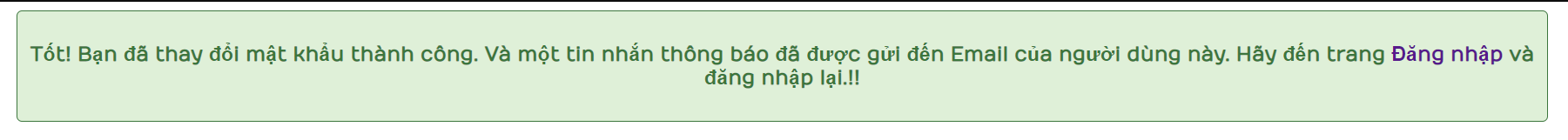
Khi người dùng nhập ô confirm password không giống với mật khẩu mới:



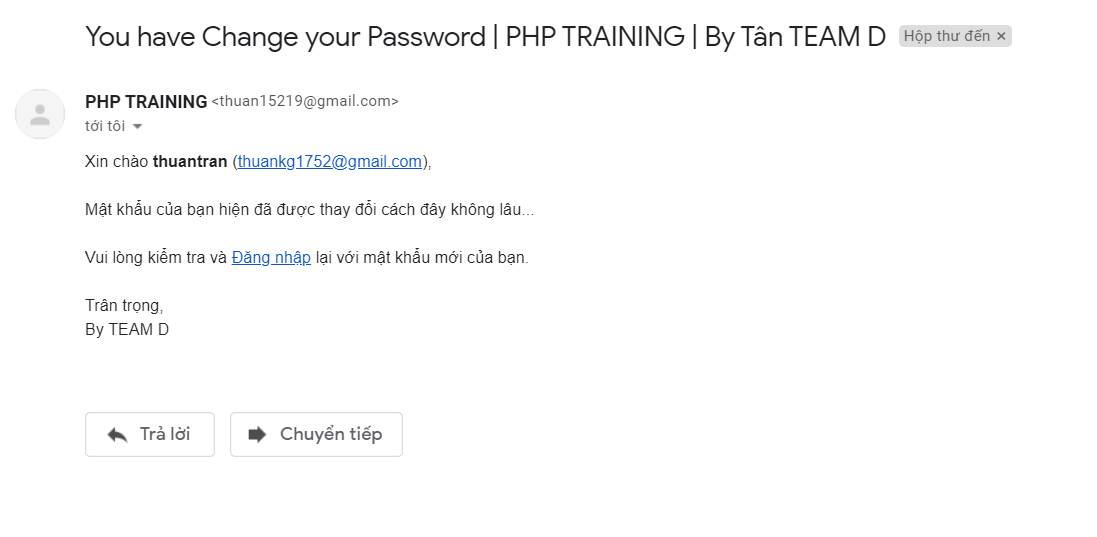
Khi người dùng nhập sai mật khẩu hiện tại:



Nếu người dùng nhập password đúng với điều kiện thỏa mãn: Có chữ cái thường, chữ in hoa, chữ số, ký tự đặc biệt và lớn hơn 8 ký tự.

Một thông báo từ trang web hiện ra với thông báo đã đổi mật khẩu và một tin nhắn trong email đã liên kết

Tin nhắn từ email

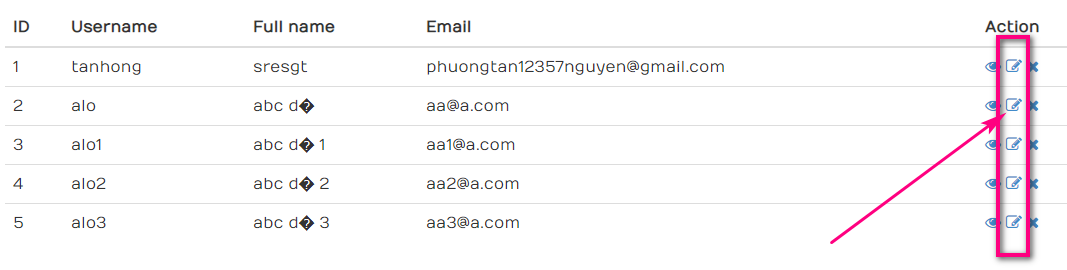


Sau khi click vào nút đăng nhập từ mail hoặc truy cập trang web để đăng nhập, lúc này người dùng sẽ phải đăng nhập với mật khẩu mới.

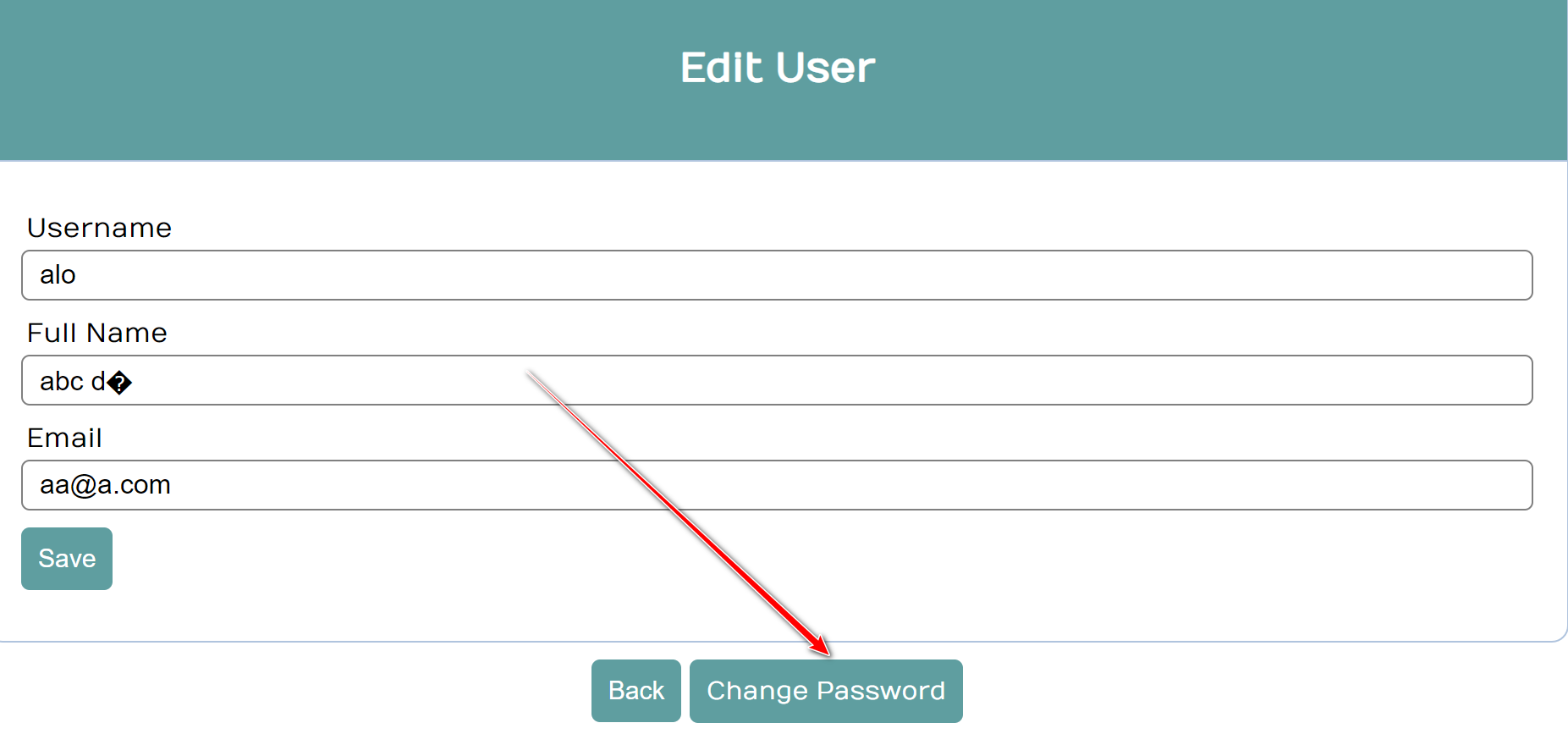
1. **Chức năng đổi mật khẩu với người dùng là Admin**

Nếu người dùng là admin sẽ cho phép sửa đổi mật khẩu của tất cả người dùng trong danh sách list.php

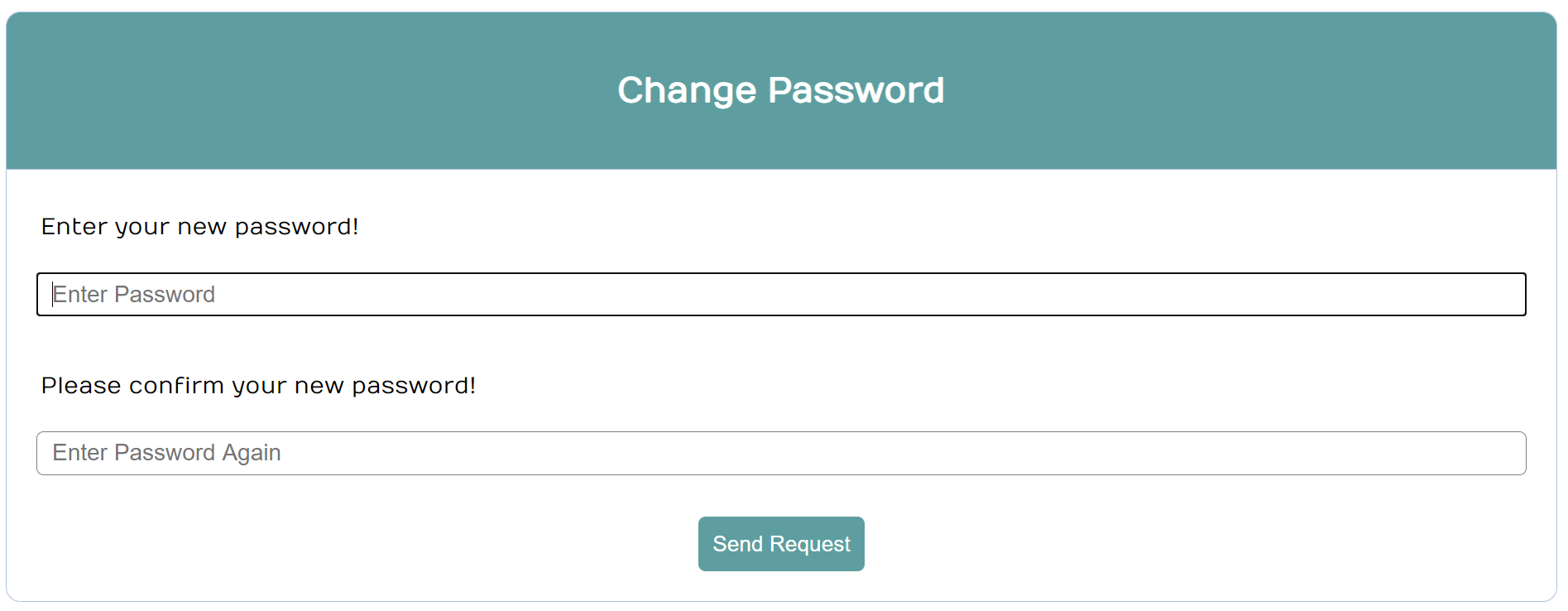
Admin nhấn vào nút có biểu tượng chỉnh sửa theo từng user.



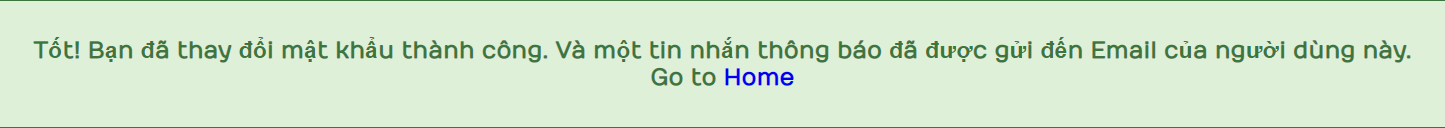
Khi vào trang chỉnh sửa theo từng user thì nhấn vào nút **Change Password.**



Đối với người dùng là admin, khi thay đổi mật khẩu cho từng người dùng thì sẽ không cần ô nhập mật khẩu hiện tại mà sẽ có quyền trực tiếp thay đổi mật khẩu mới cho user khác.



Sau khi thay đổi mật khẩu, xác nhận đúng mật khẩu và nhấn **Send Request** thì sẽ thấy thông báo sau:

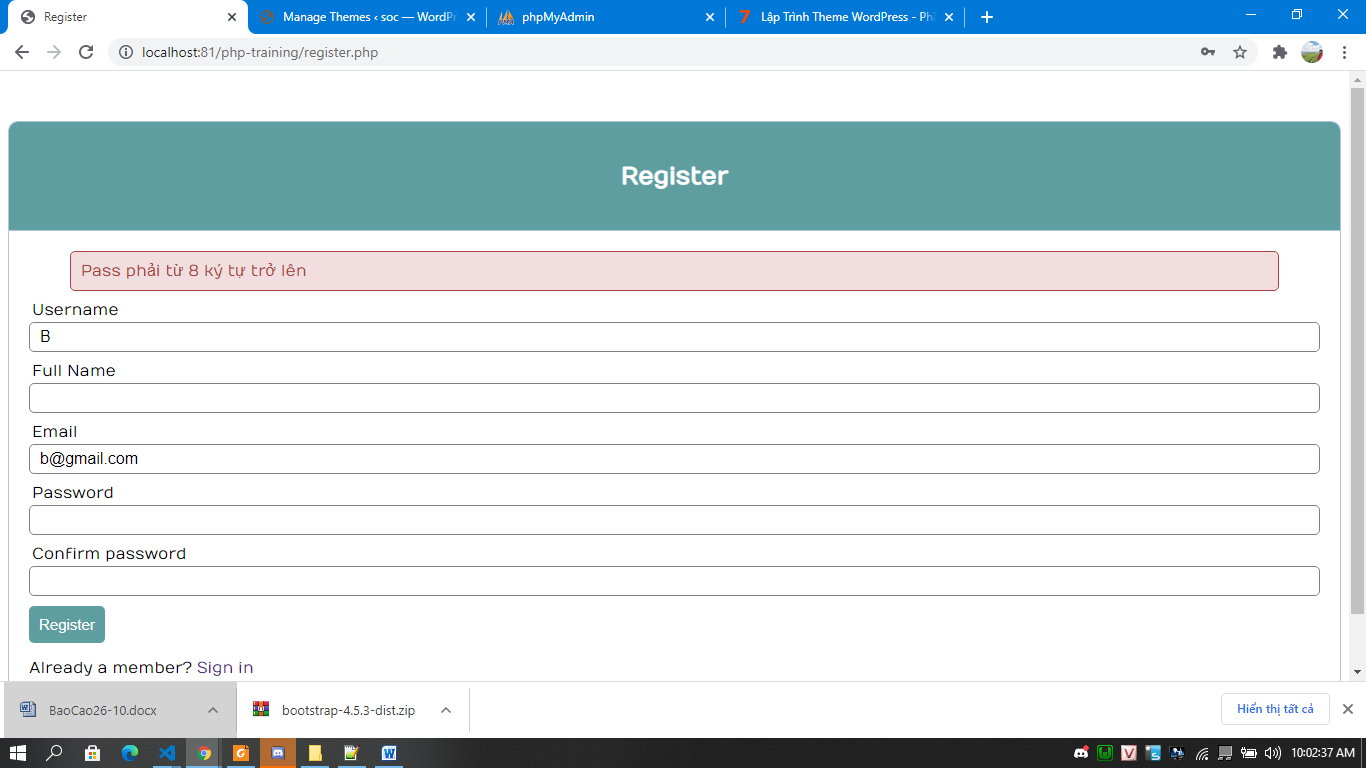


Email thông báo đã đổi mật khẩu mới sẽ gửi đến Email mà người dùng này đã đăng ký trước đó!

Đối với các user khác thì thao tác cũng làm tương tự.

1. **Ràng buộc mật khẩu khi đăng ký phải có mức độ bảo mật cao**

Trong form Register khi người dùng nhập password nếu password không đủ 8 kí tự sẽ báo lỗi



Chúng ta sẽ viết điều kiện khi người dùng nhập vào

$password\_1 = escape($\_POST['password\_1']);

elseif (!preg\_match("/(?=.\*[0-9])(?=.\*[A-Z])(?=.\*[a-z]).{8,}/", $password\_1)){

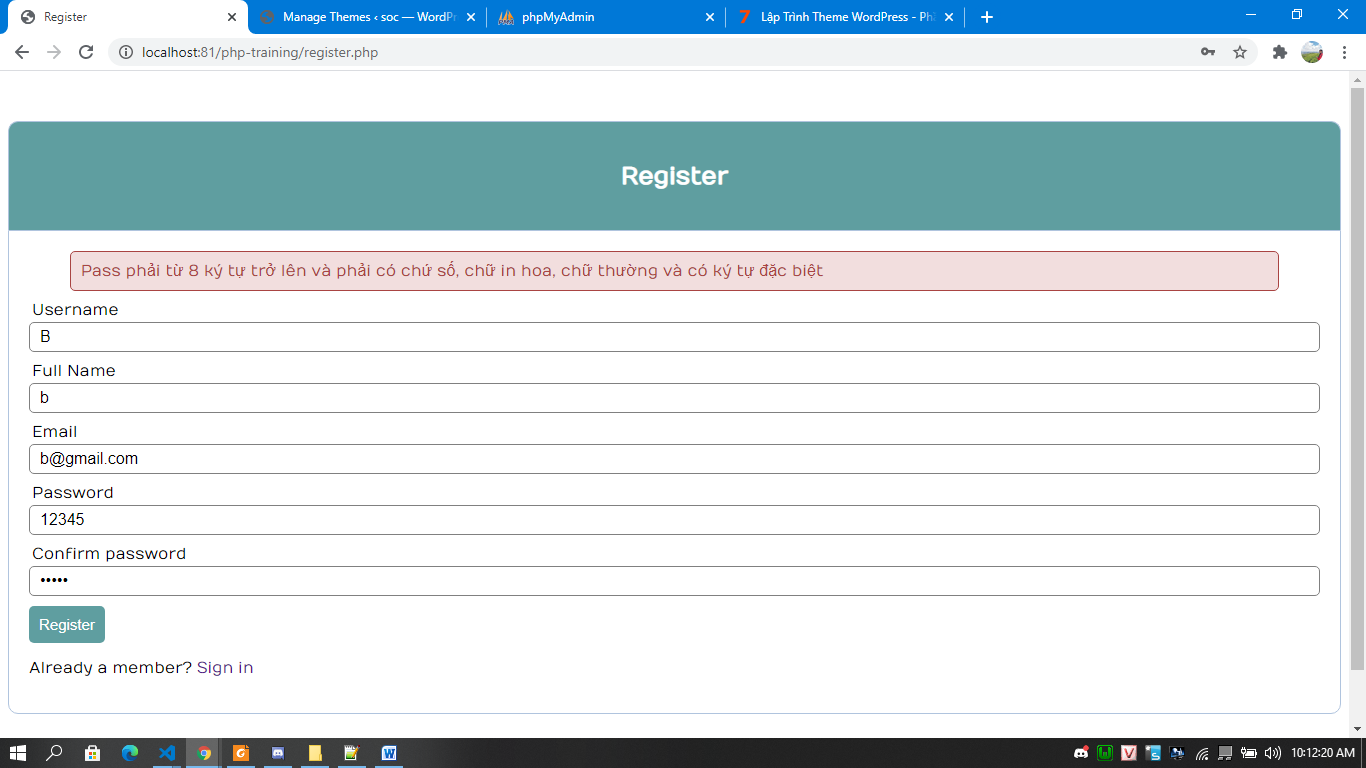
        array\_push($errors, "Pass phải từ 8 ký tự trở lên ");

    }

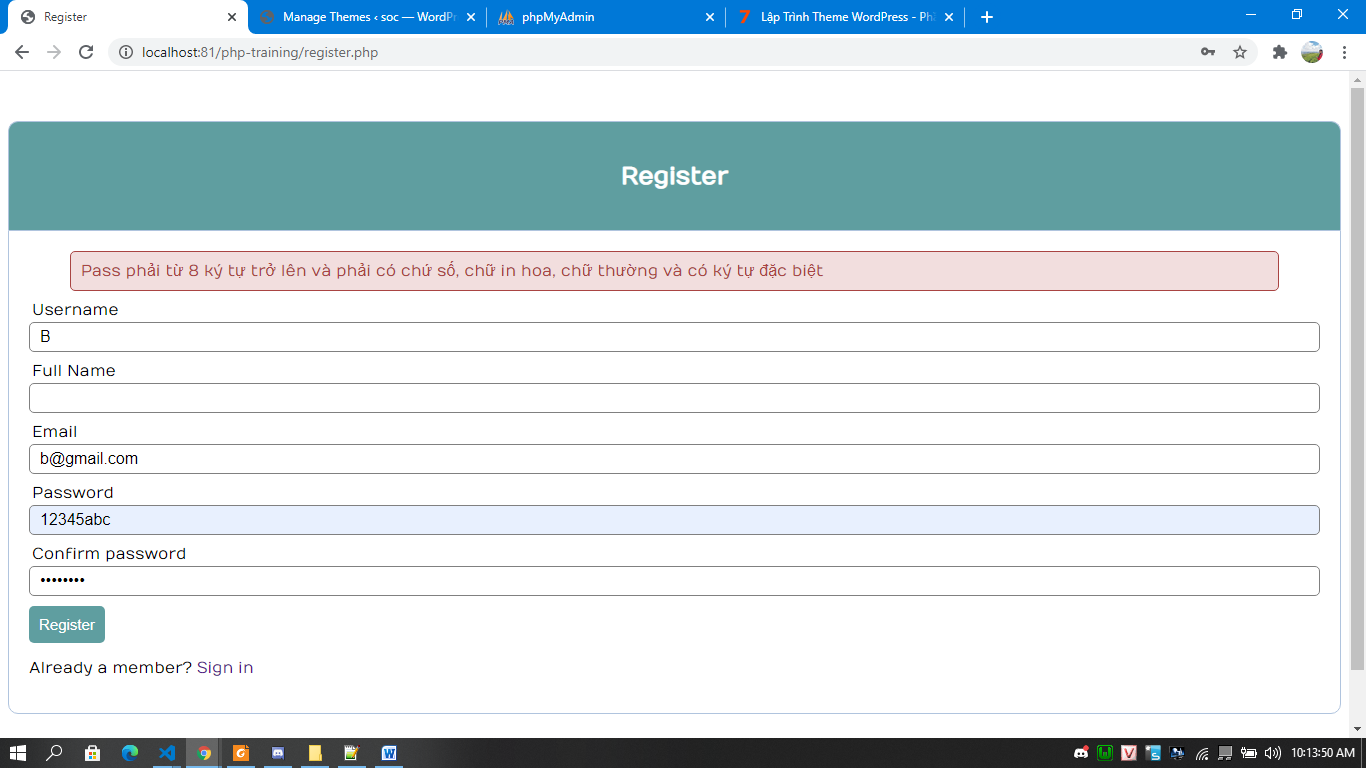
Trong điều kiện này: độ dài phải lớn hơn 8 kí tự, và ít nhất 1 kí tự số từ 0 đến 9 , ít nhất 1 kí tự thường từ a đến z, ít nhất 1 kí tự in hoa từ A đến Z.

**Đây là ví dụ demo:**

**Trường hợp 1**: pass: 12345



**Trường hợp 2**: pass: 12345abc



**Trường hợp 3**: pass: 12345ABC

