**Bài tập thực hành 04**

Lập trình PHP nâng cao

**Mục tiêu cần đạt**:

Sau buổi các bạn cần nắm thao tác sử dụng PHP nâng cao

* Hiểu và biết cách dùng Cookies và Session
* Tạo form đăng nhập và dùng cookies, session để quản lý các đăng nhập, thực hiện chức năng log out (thoát) khỏi hệ thống
* Biết về SQL Injection.
* Upload file, lưu thông tin về tập tin trong CSDL
* Đọc, và xử lý nội dung trong tập tin

Các bạn đọc và thực hiện viết code như hướng dẫn để làm quen, sau đó đọc trả lời các **Yêu cầu** ở dưới bài hướng dẫn để thực hiện yêu cầu bài thực hành.

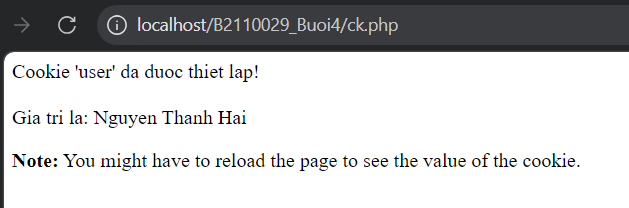
**Gợi ý Tham khảo:**

* <https://www.w3schools.com/php/>
* và các nguồn khác mà sinh viên tìm được

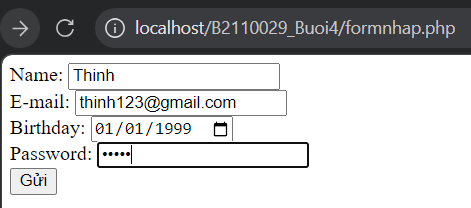
**Yêu cầu bài thực hành:**

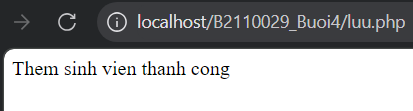
# 1. Bạn hãy chạy tất cả các lệnh hướng dẫn ở trên và chụp lại màn hình kết quả.

**Tạo Cookies**



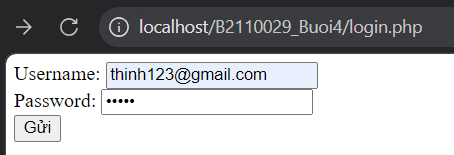
**Tạo 1 trang đăng ký người dùng với mật khẩu được mã hóa**

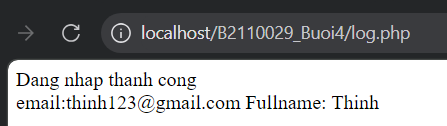




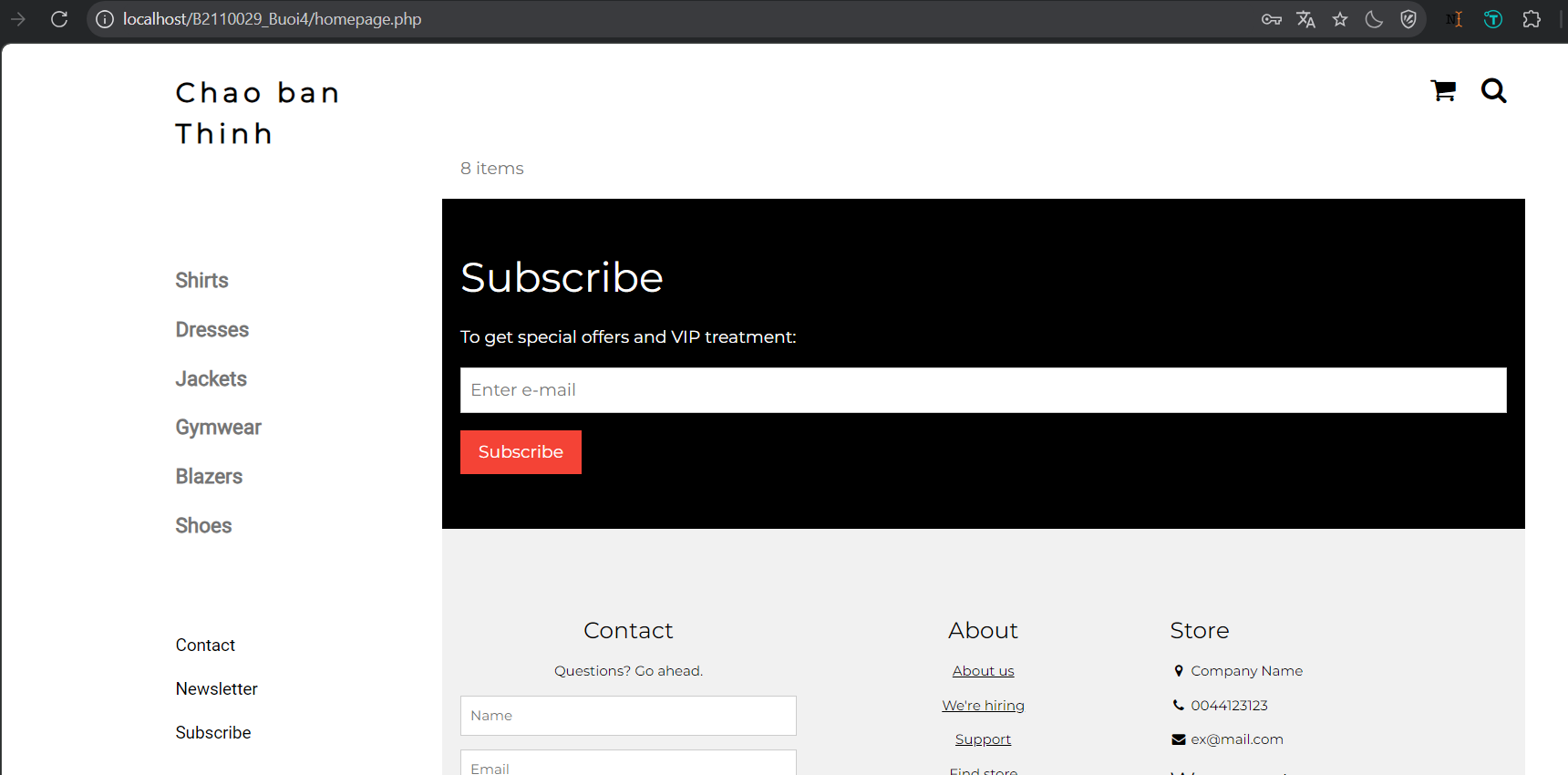


**Tạo form đăng nhập**

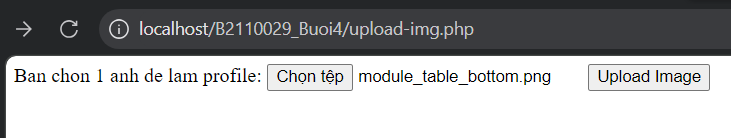


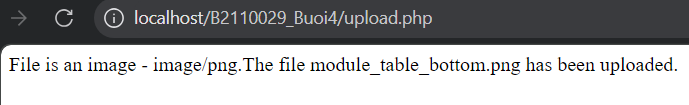


**Dùng Cookies lưu lại tên đăng nhập, email và điều hướng**

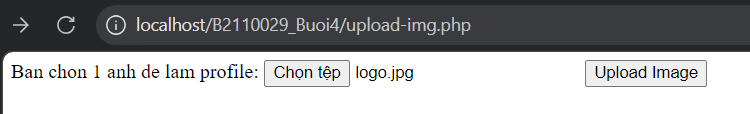


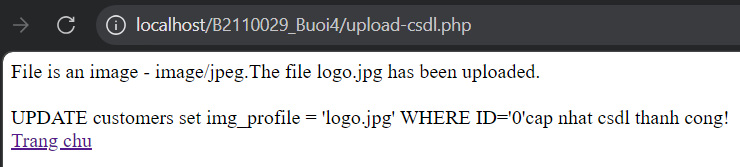
**Upload tập tin**

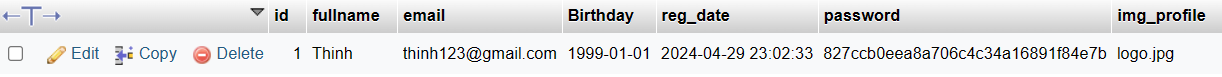


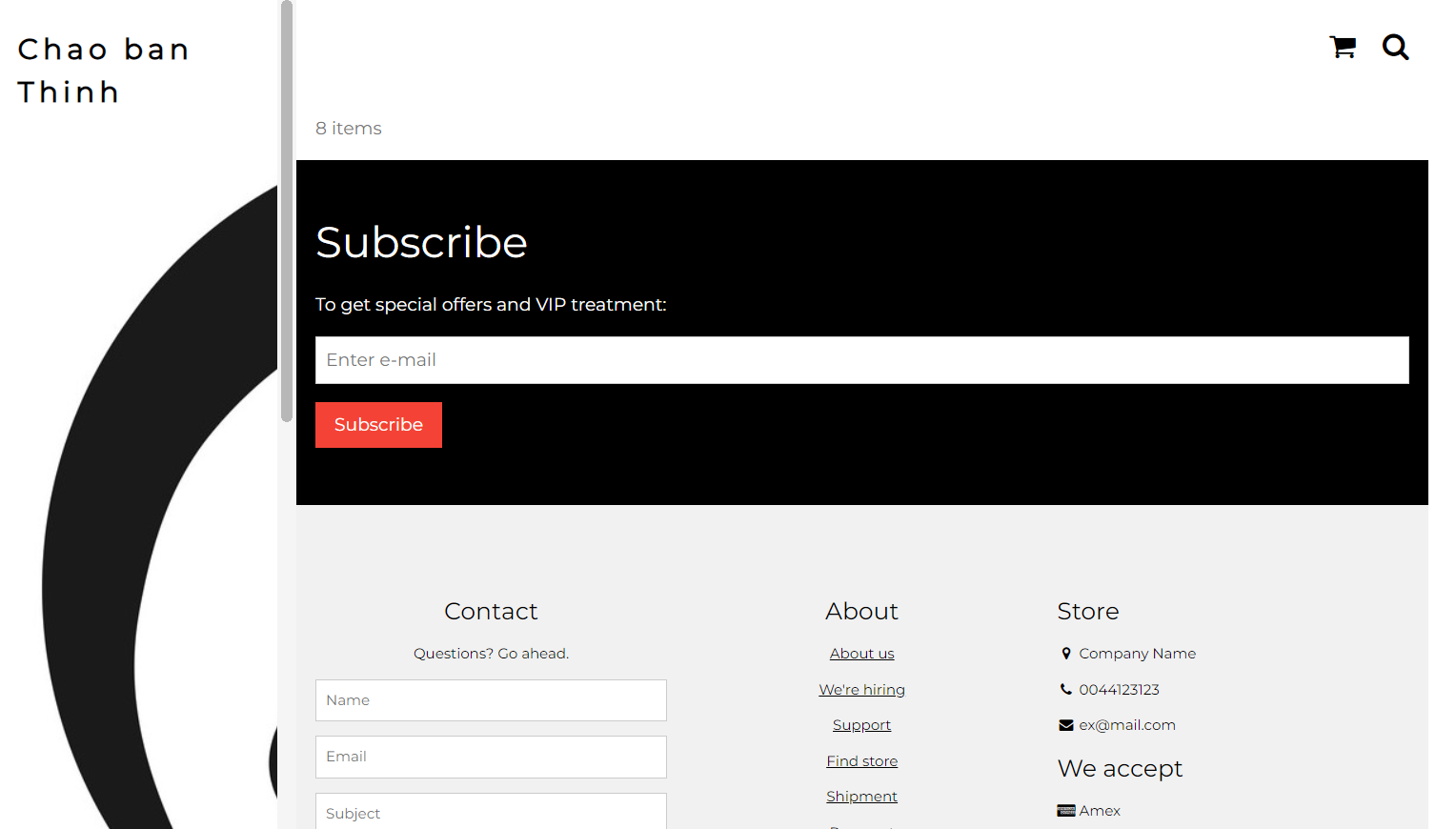


**Upload ảnh và lưu dữ liệu về ảnh trong CSDL**

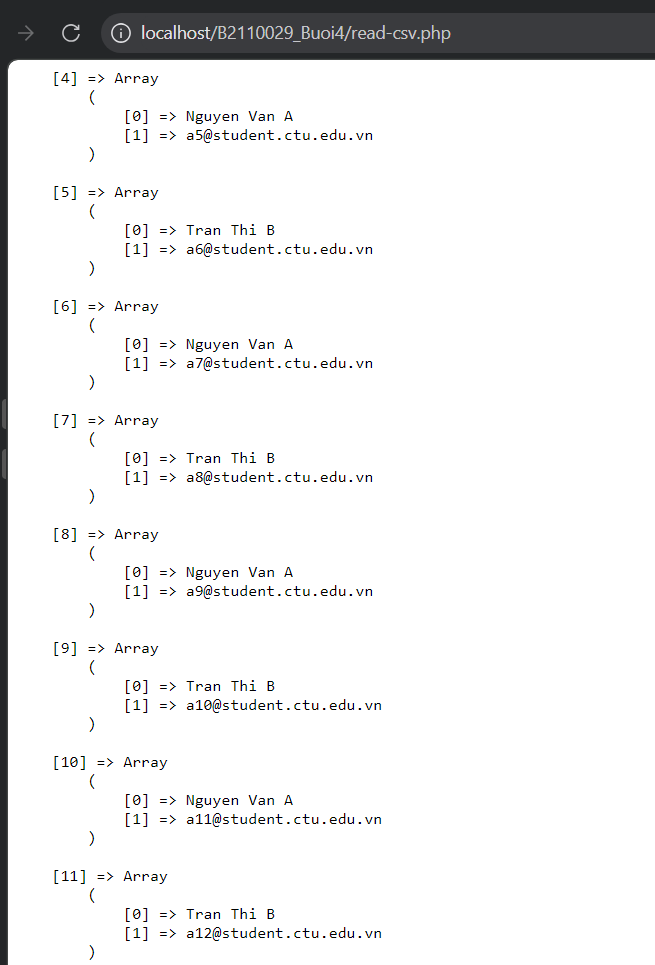
****

****

****

****

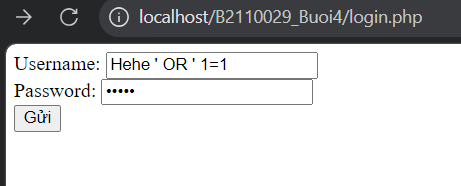
**Upload file csv và đọc dữ liệu vào mảng**

****

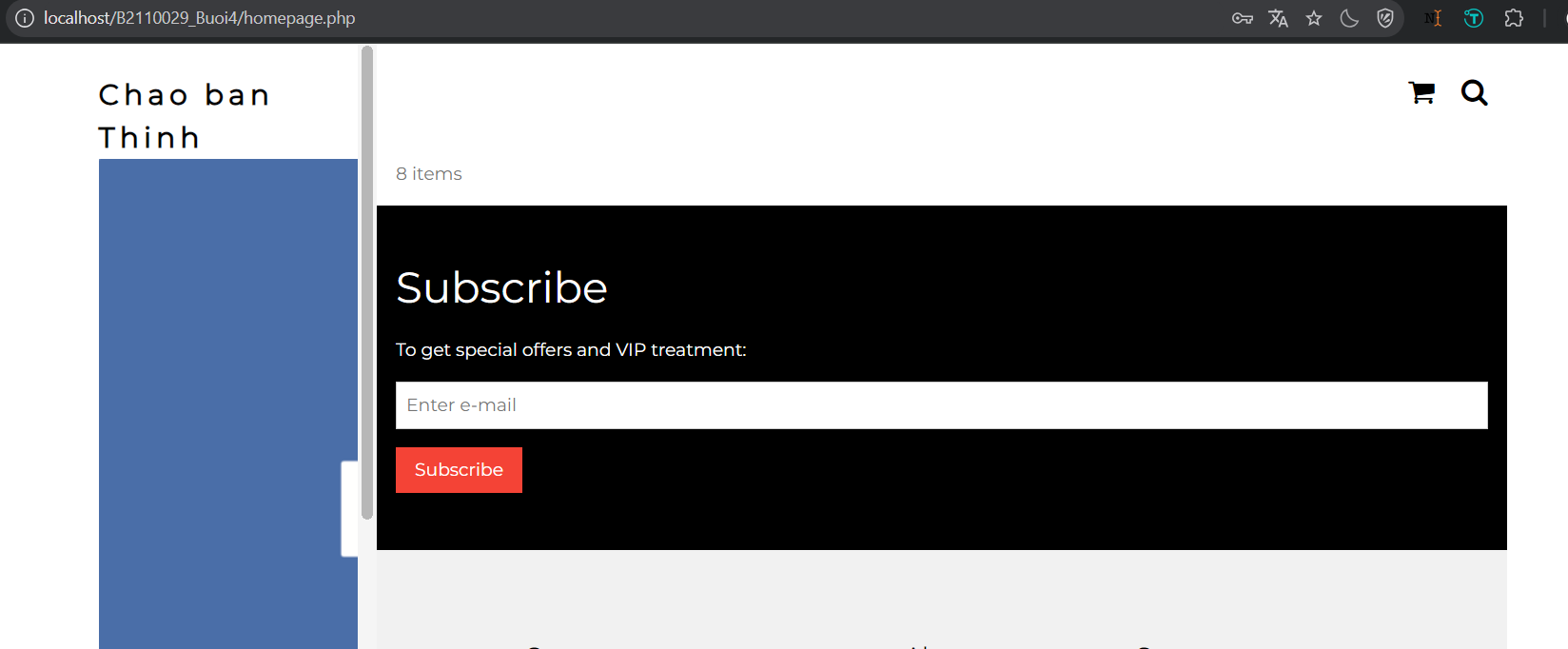
# 2. Trong vài trường hợp, hacker có thể sử dụng các kỹ thuật SQL Injection để hack hệ thống của bạn. Bạn hãy trình bày SQL Injection là gì và thử các ví dụ trình bày trong . Ứng dụng kỹ thuật đó vào trang đăng nhập mà bạn đã tạo, chụp lại các kết quả.

- SQL injection là một kỹ thuật cho phép những kẻ tấn công lợi dụng lỗ hổng của việc kiểm tra dữ liệu đầu vào để inject và thi hành các câu lệnh SQL bất hợp pháp.

Ví dụ: Khi ta đăng nhập vào tài khoản [thinh123@gmail.com](mailto:thinh123@gmail.com) (có mật khẩu là 12345)

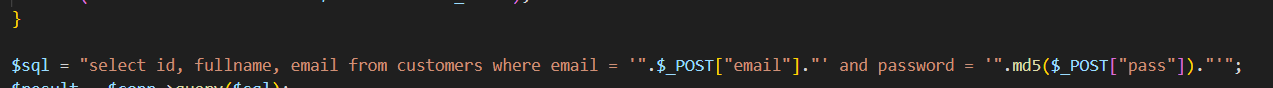


Ta sẽ để username là Hehe ' OR ' 1=1, kết quả chạy như sau :



🡪Giải thích cơ chế hack :

Thực chất, khi ta đăng nhập ở trang login.php, khi đăng nhập xong sẽ chuyển qua trang log.php. Tại đây sẽ xác nhận tài khoản và mật khẩu nhập vào rồi mới quyết định việc có cho người dùng vào hệ thống hay không. Tuy nhiên, trong code của log.php có đoạn như sau :



Hacker có thể lợi dụng kẻ hở nơi username nhập vào và chèn đoạn « Hehe ' OR ' 1=1 »

Lúc này, câu lệnh truy vấn sẽ bị đổi thành

« select id, fullname, email from customers where email = *Hehe OR 1=1* and password = 12345.

Vì 1=1 luôn là kết quả TRUE nên câu lệnh truy vấn trên luôn hợp lệ=>Đăng nhập vào hệ thống thành công.

# 3. Dựa vào link2, bạn hãy cho biết Cookie là gì, diễn giải ý nghĩa các tham số trong setcookie(). Cách lưu, lấy, xóa giá trị trong cookie.

- Cookie thường được sử dụng để xác định người dùng. Cookie là một tệp nhỏ mà máy chủ nhúng vào máy tính của người dùng.

setcookie(name, value, expire, path, domain, secure, httponly): name là bắt buộc, các tham số khác là tùy chọn

- Diễn giải các tham số trong hàm setcookie:

+ name: tên.

+ value: giá trị.

+ expire: thời hạn.

+ path: đường dẫn.

+ domain: tên miền.

+ secure: true nếu sử dụng HTTPS để truyền dữ liệu (mặc định false).

+ httponly: mặc định là false, nếu true thì chỉ cho phép cookie với giao thức HTTP.

- Lưu: từ PHP có thể thiết lập Cookie bằng hàm setcookie (phải xuất hiện trước thẻ html).

- Lấy: dùng biến $\_COOKIE để truy cập thông tin cookie (có thể dùng isset() để kiểm tra sự tồn tại của COOKIE).

- Xóa: sử dụng chính hàm setcookie với thủ thuật đặt thời gian đã hết hạn.

VD: setcookie( "$cookie\_name", "$cookie\_value", time() - 60, "/","", 0);

# 4. Dựa vào link, bạn hãy cho biết Session dùng để làm gì. Cách lưu, lấy, xóa giá trị trong Session.

- Session dùng để lưu trữ thông tin (trong các biến) sẽ được sử dụng trên nhiều trang. Không giống như cookie, thông tin không được lưu trữ trên máy tính của người dùng.

- Lưu: một phiên được bắt đầu với session\_start() chức năng. Các biến session được đặt bằng biến toàn cục PHP: $\_SESSION.

- Lấy: dùng biến $\_SESSION để truy cập thông tin session.

- Xóa: để xóa session sử dụng hàm session\_unset() và session\_destroy() .

# 5. Bạn hãy so sánh Cookie và Session.

- Về Cookie:

+ Cookie được lưu trữ trên trình duyệt người dùng.

+ Cookie dược lưu trữ ở phía máy khách.

+ Cookie dễ dàng sửa đổi vì được lưu trữ ở phía máy khách.

+ Dữ liệu có sẵn trong trình duyệt chạy đến khi hết hạn.

- Về Session:

+ Session không được lưu trữ trên trình duyệt người dùng.

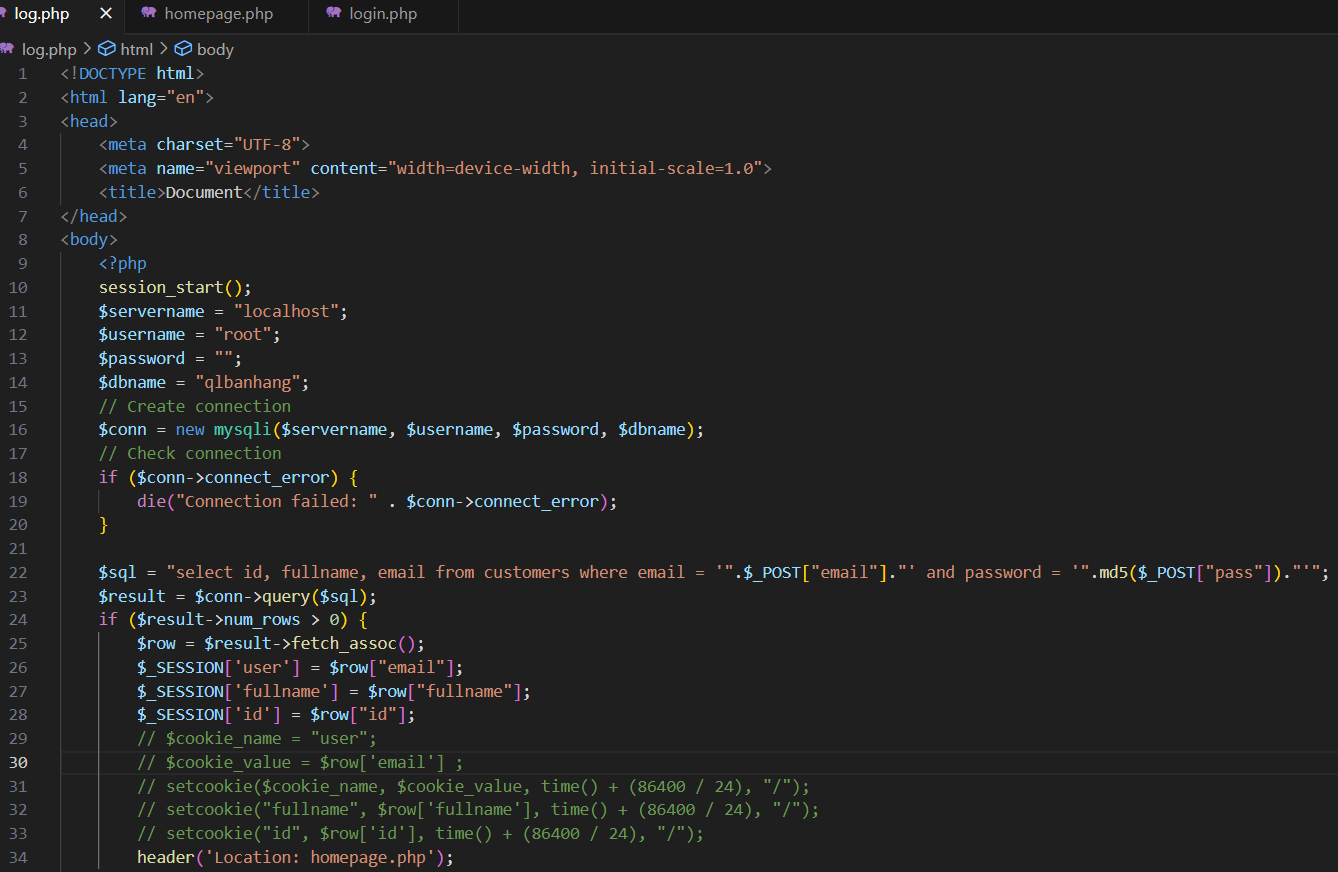
+ Session được lưu trữ ở phía máy chủ.

+ Session không dễ dàng sửa đổi vì được lưu trữ ở phía máy chủ.

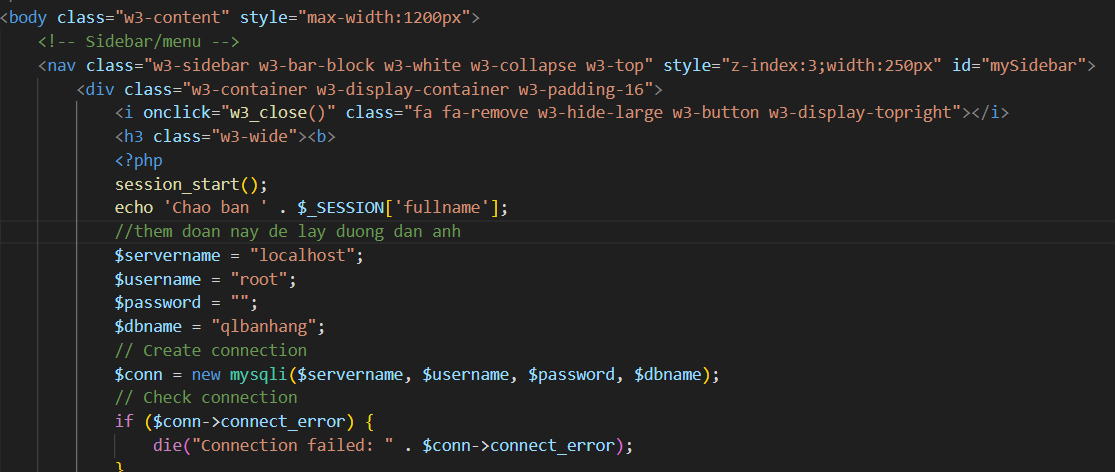
+ Dữ liệu có sẵn cho trình duyệt chạy, sẽ mất thông tin session khi đóng trình duyệt.

# 6. Chỉnh sửa các đoạn gán, khởi tạo, lấy giá trị cookie trong log.php và homepage.php thay bằng cách dùng Session. Tạo tập tin thoat.php để xóa các giá trị trong session, cookie thực hiện chức năng log out khỏi hệ thống.

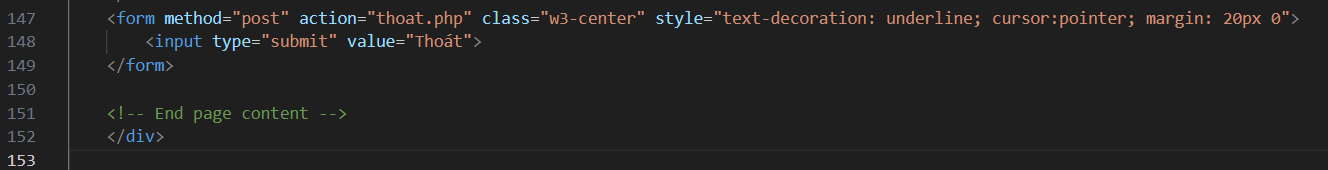
- Sửa trong log.php:



- Sửa trong homepage.php:



- Cũng trong homepage.php, tiến hành thêm dòng html với thẻ form sau để có chức năng thoát khỏi trang web

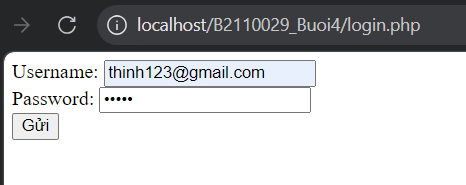


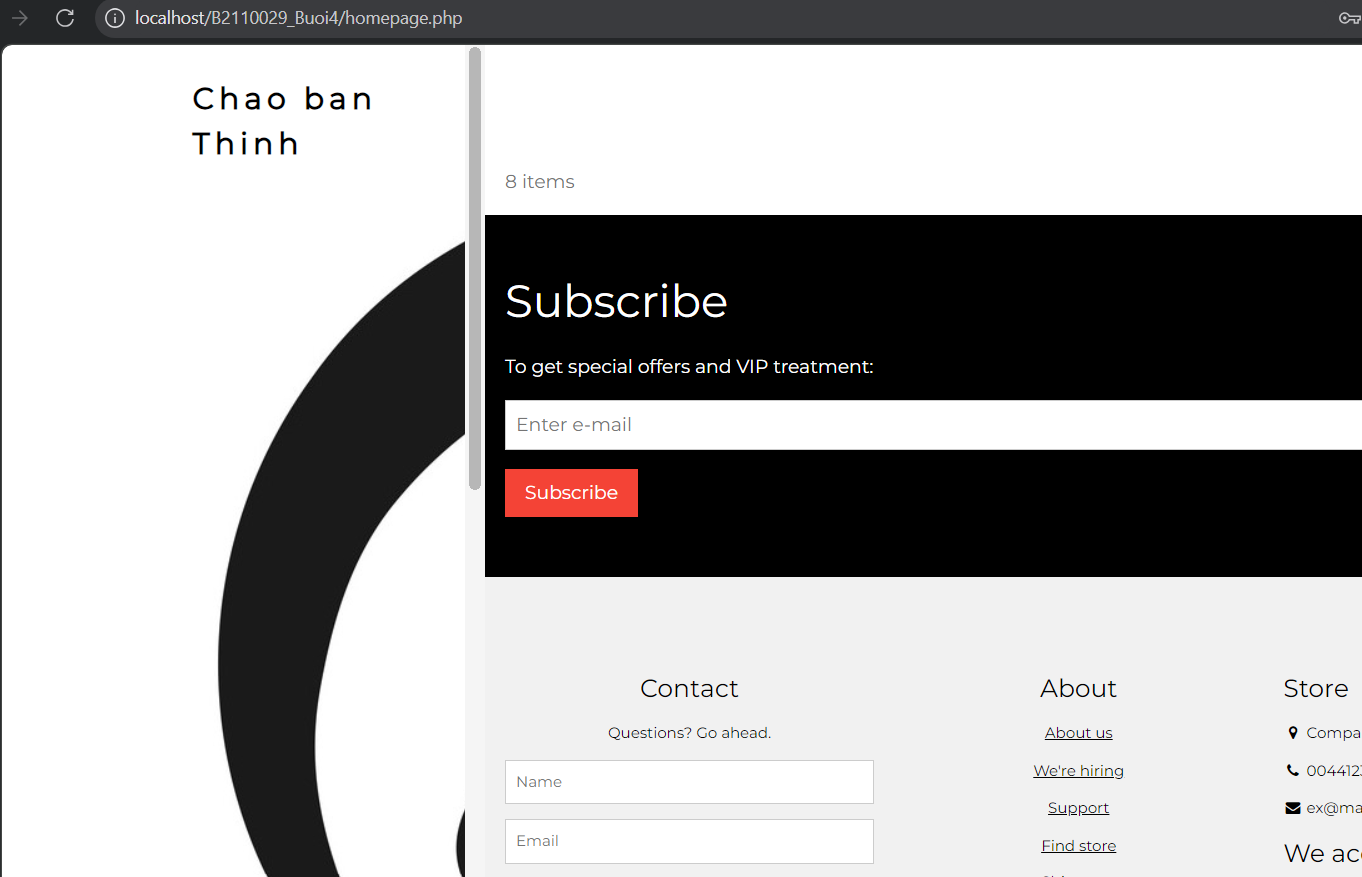
- Tạo thoat.php:



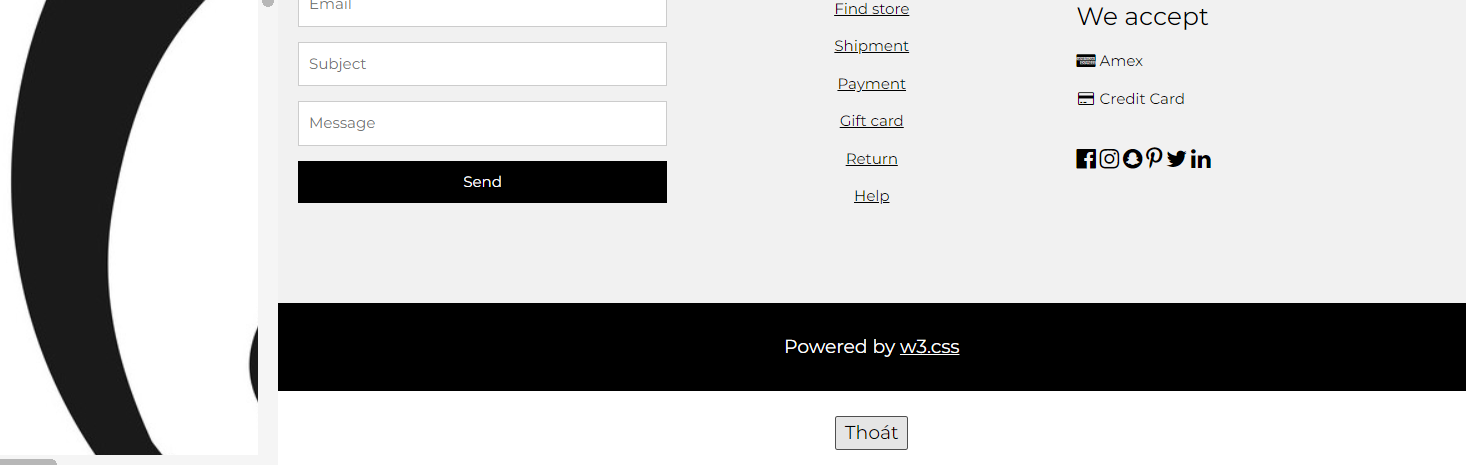
- Tiến hành trải nghiệm các chức năng

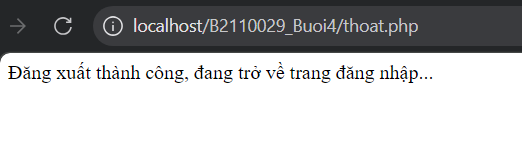
+ Đăng nhập thử





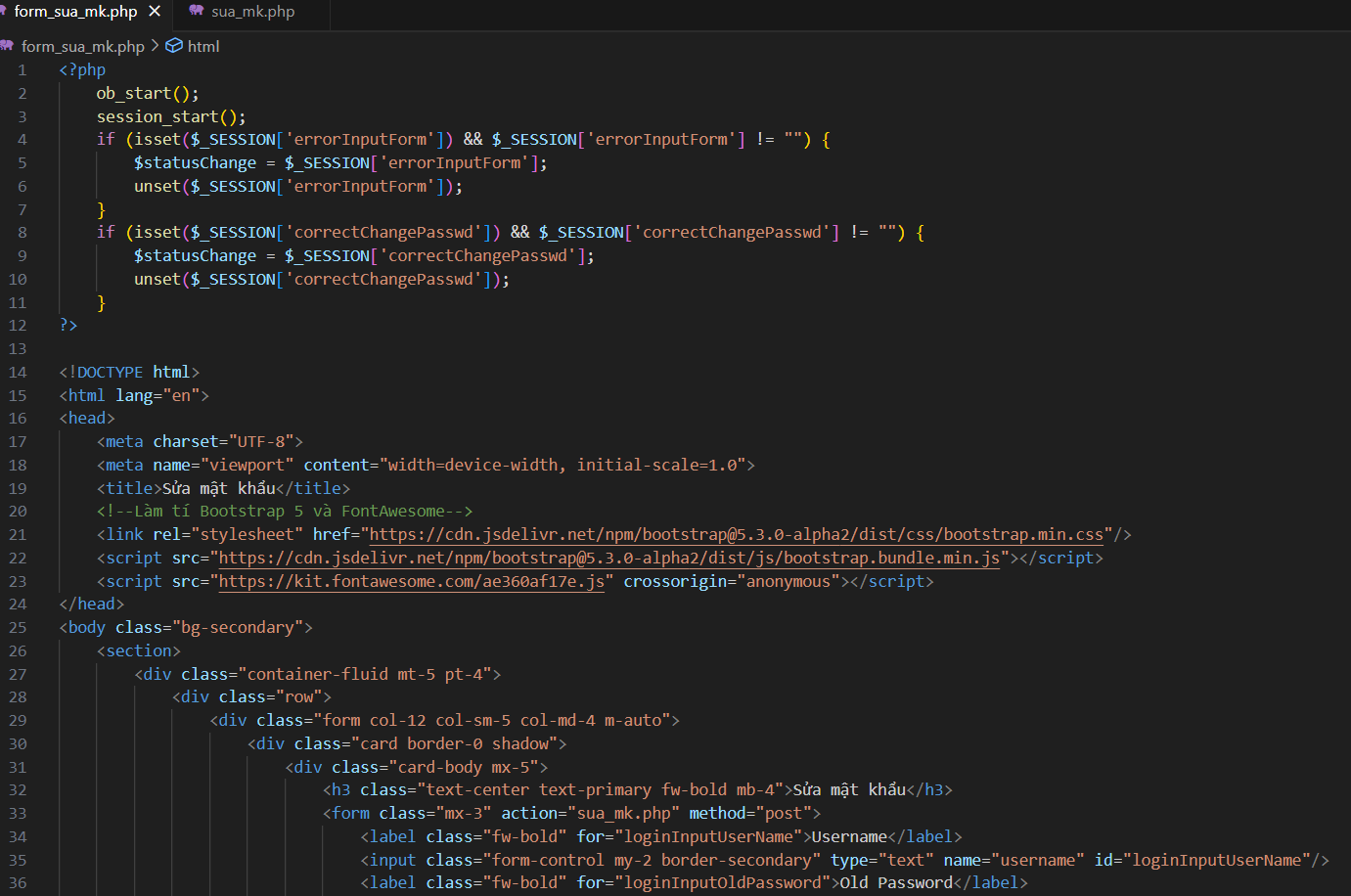
+ Thử thoát

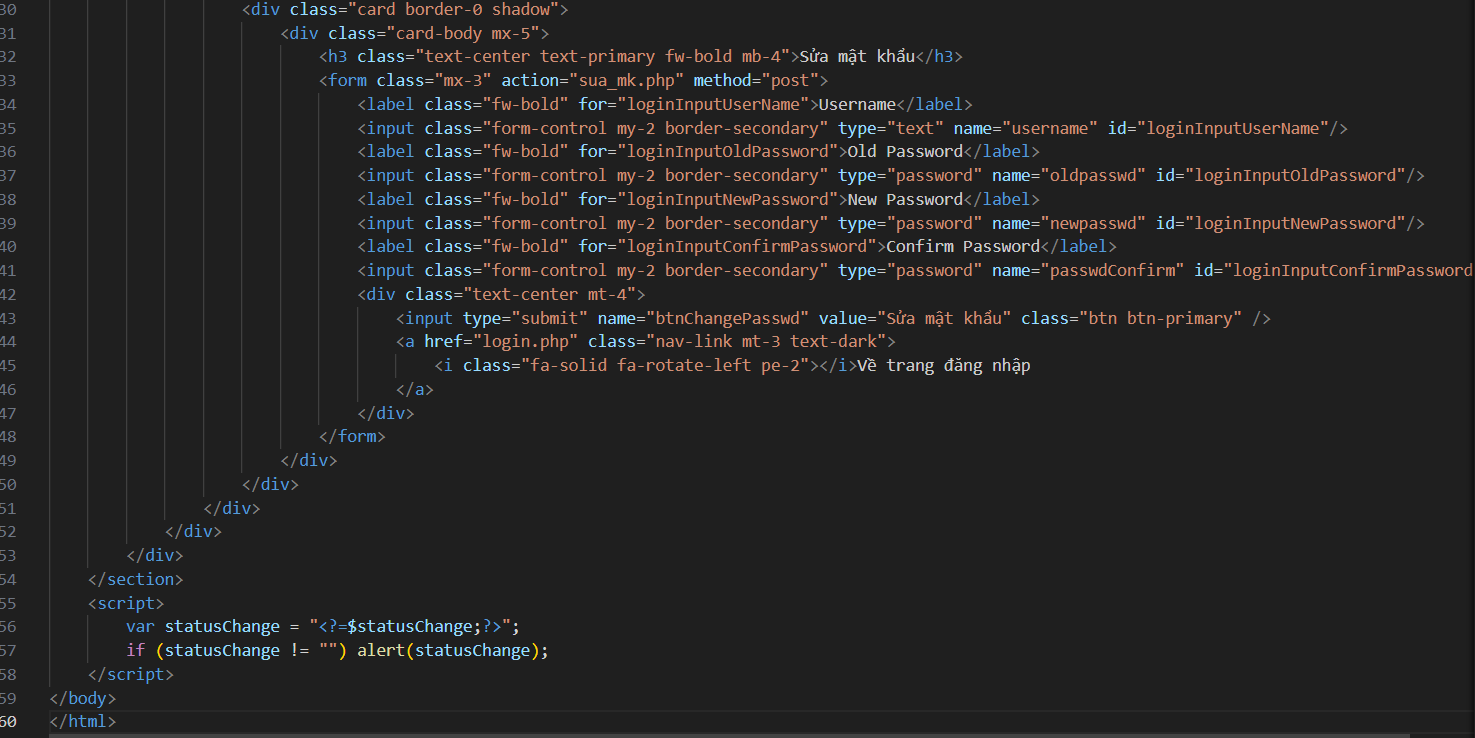




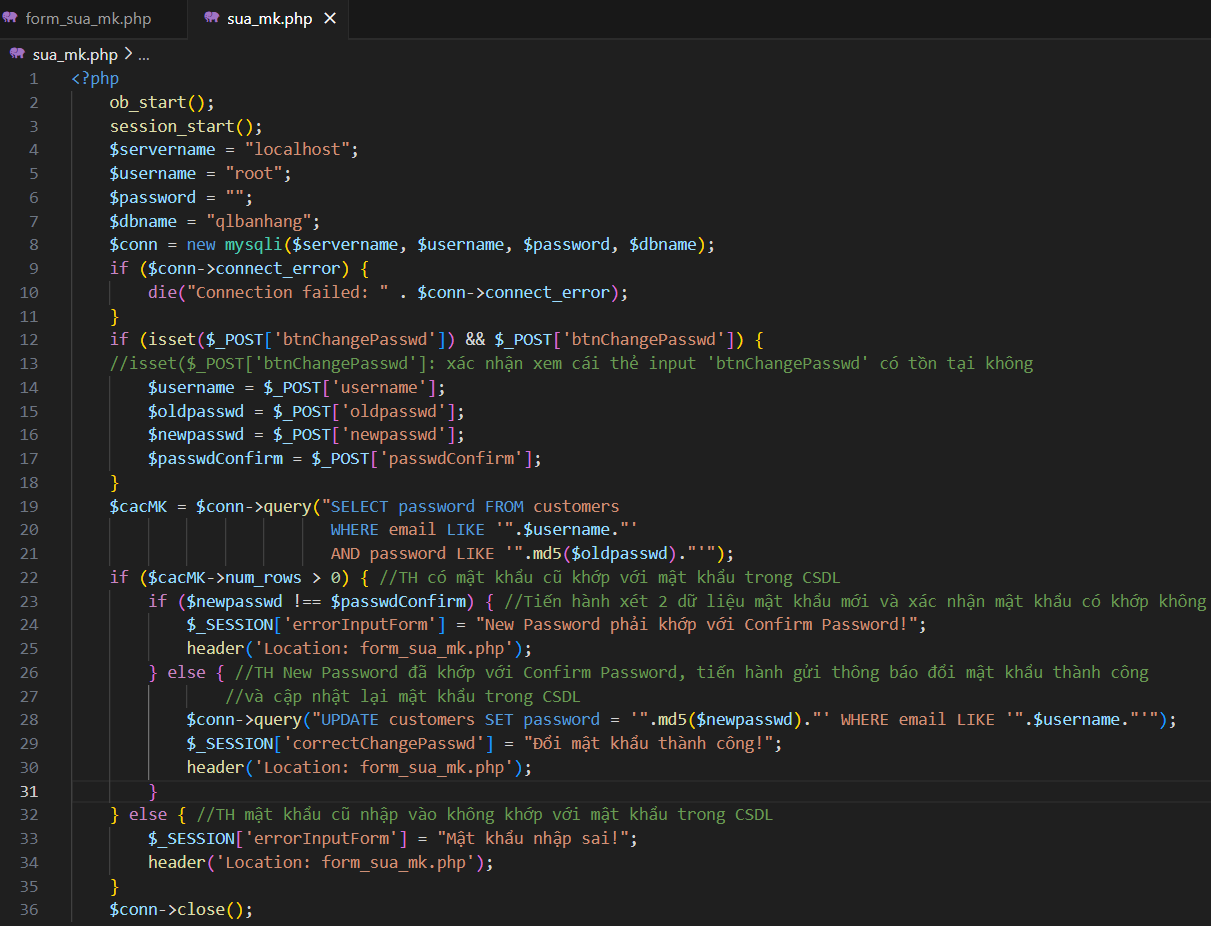
# 7. Bạn hãy tạo *form\_sua\_mk.php* cho phép người dùng chỉnh sửa mật khẩu sau khi đăng nhập. Yêu cầu gồm 3 input: ô để nhập mật khẩu cũ, 1 ô để nhập mật khẩu mới, 1 ô cho phép nhập mật lại mật khẩu mới. Khi nhập xong cần đảm bảo: mật khẩu cũ là khớp với CSDL đang lưu, 2 ô nhập mật khẩu mới phải khớp với nhau và không giống với mật khẩu cũ, nếu đáp ứng điều kiện thì tiến hành băm mật khẩu với md5 và lưu mật khẩu mới vào CSDL.

- Ta tiến hành tạo file form\_sua\_mk.php, file này sẽ là 1 form yêu cầu người dùng nhập vào, sau đó sẽ gửi dữ liệu qua sua\_mk.php để xử lý



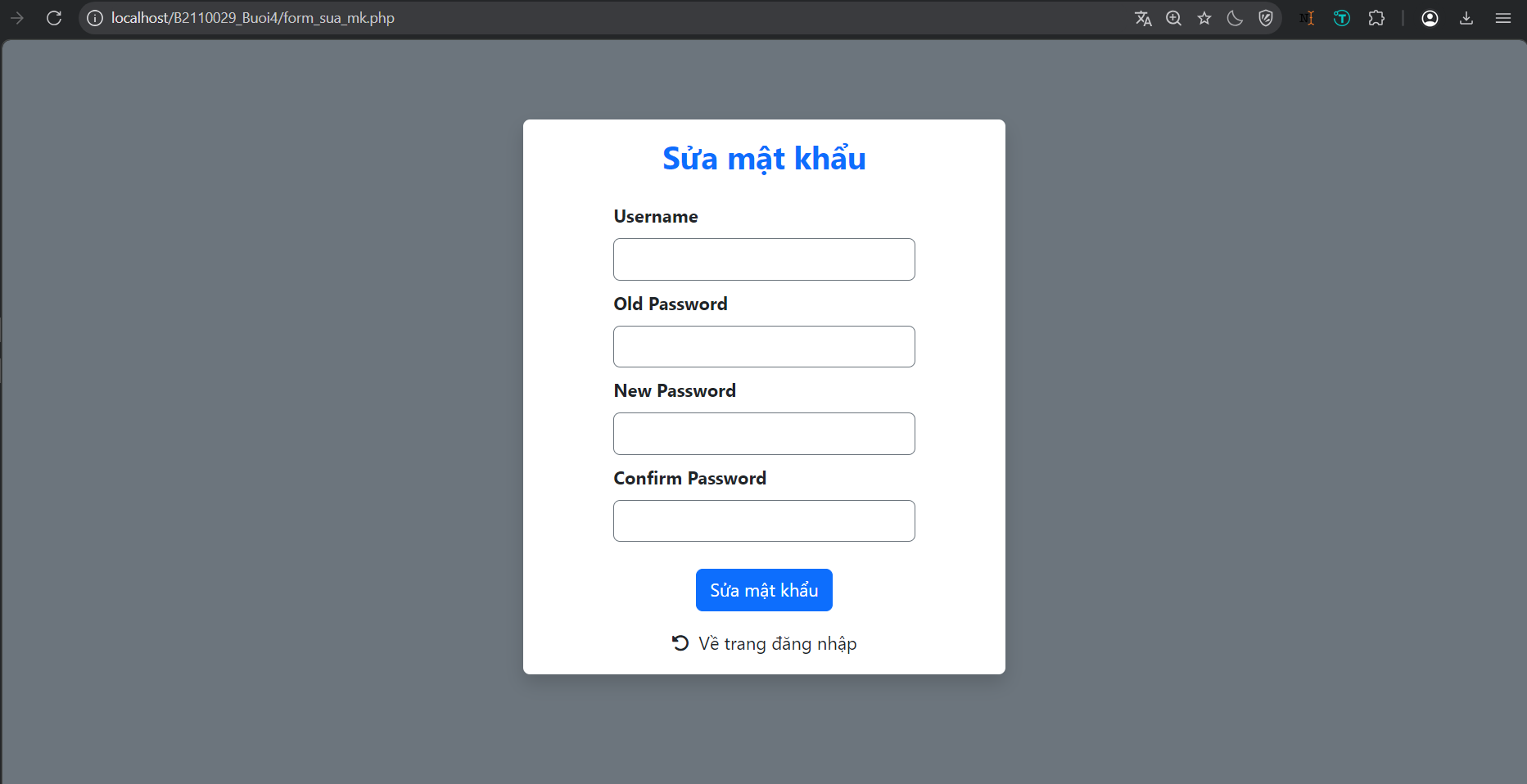


- Tiến hành tạo file sua\_mk.php để xử lý các input mà người dùng nhập vào

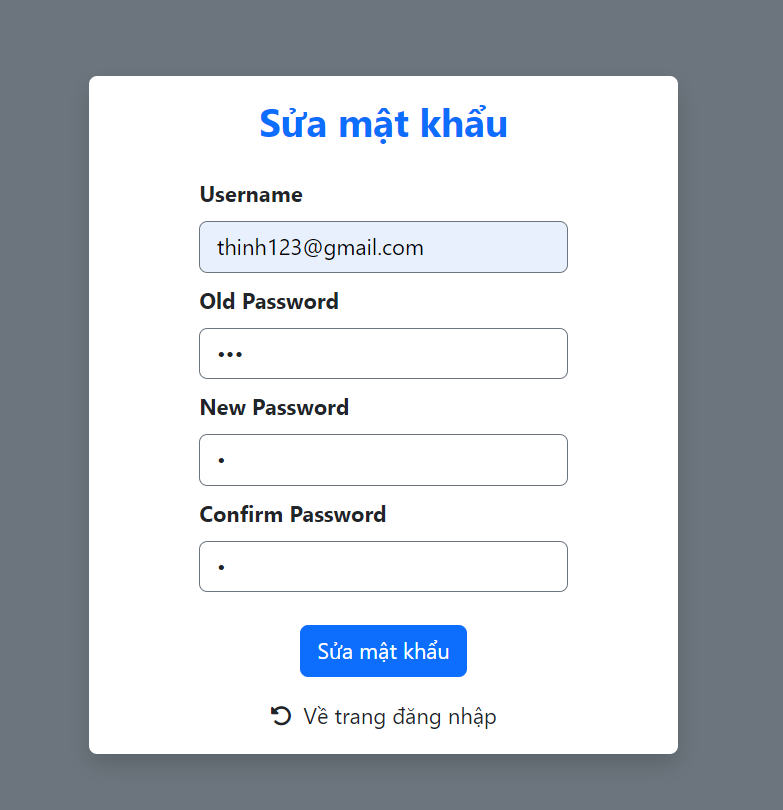


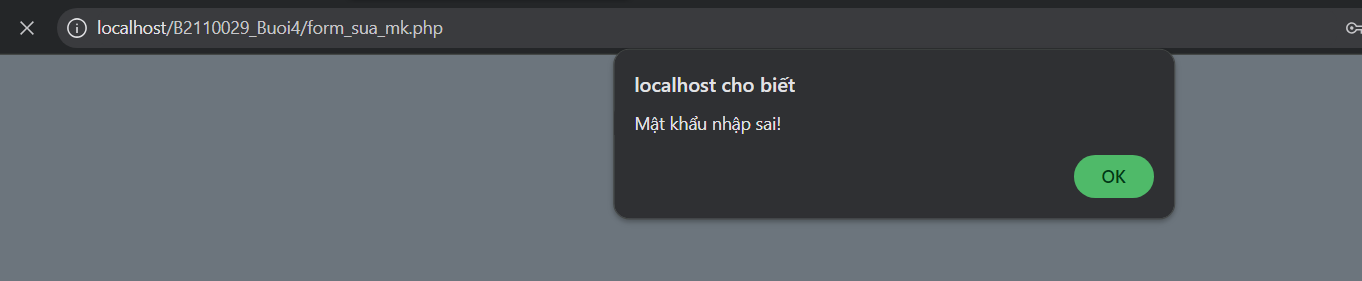
- Tiến hành chạy các trang thử

+ form\_sua\_mk.php

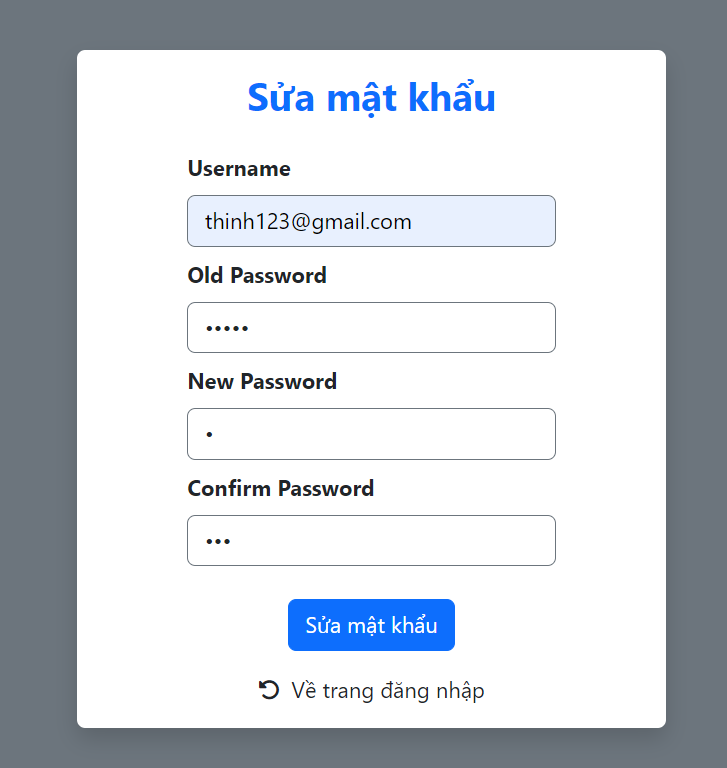


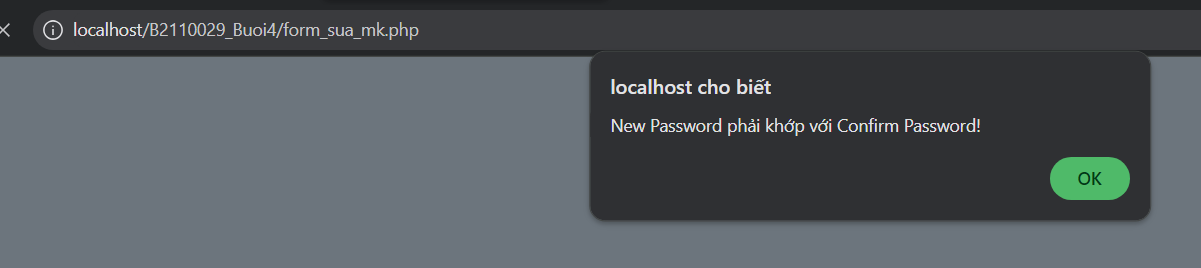
* Trường hợp nhập mật khẩu sai với mật khẩu trong CSDL



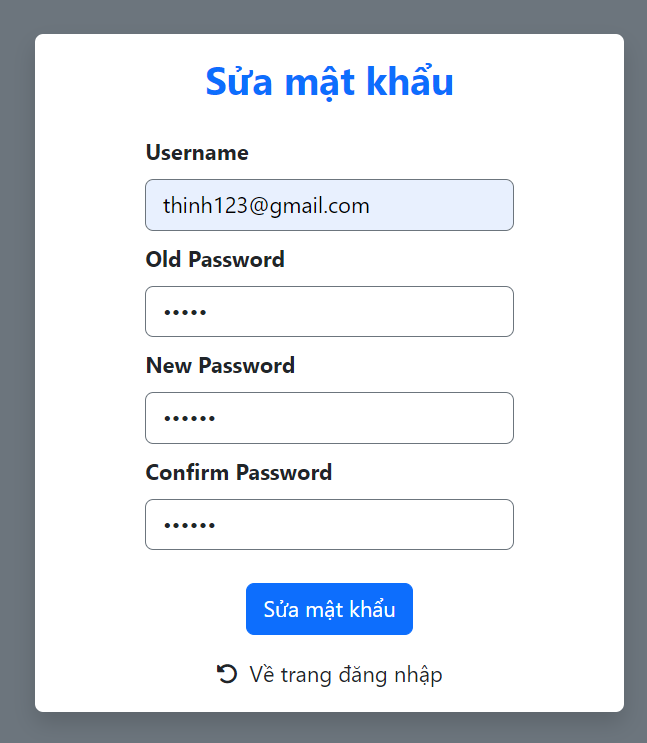


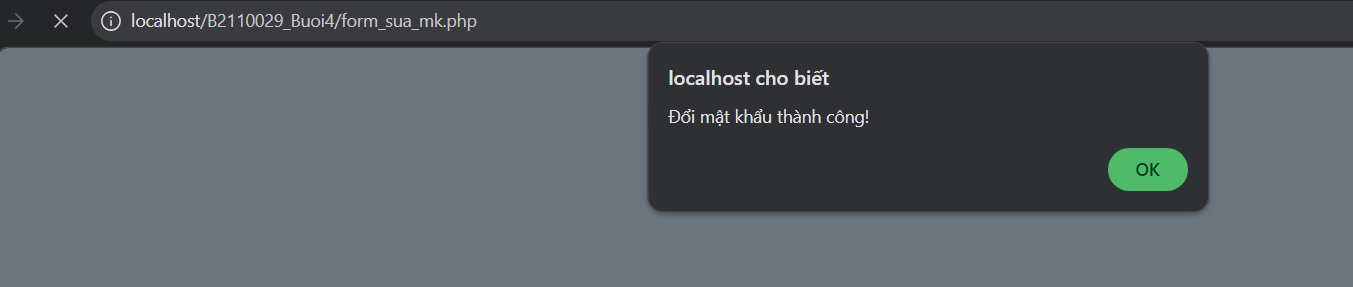
* Trường hợp mật khẩu mới không khớp với xác nhận mật khẩu





* Trường hợp tất cả đều hợp lệ, một thông báo thay đổi mật khẩu thành công sẽ hiện ra

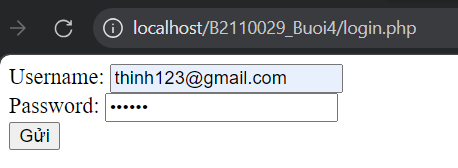


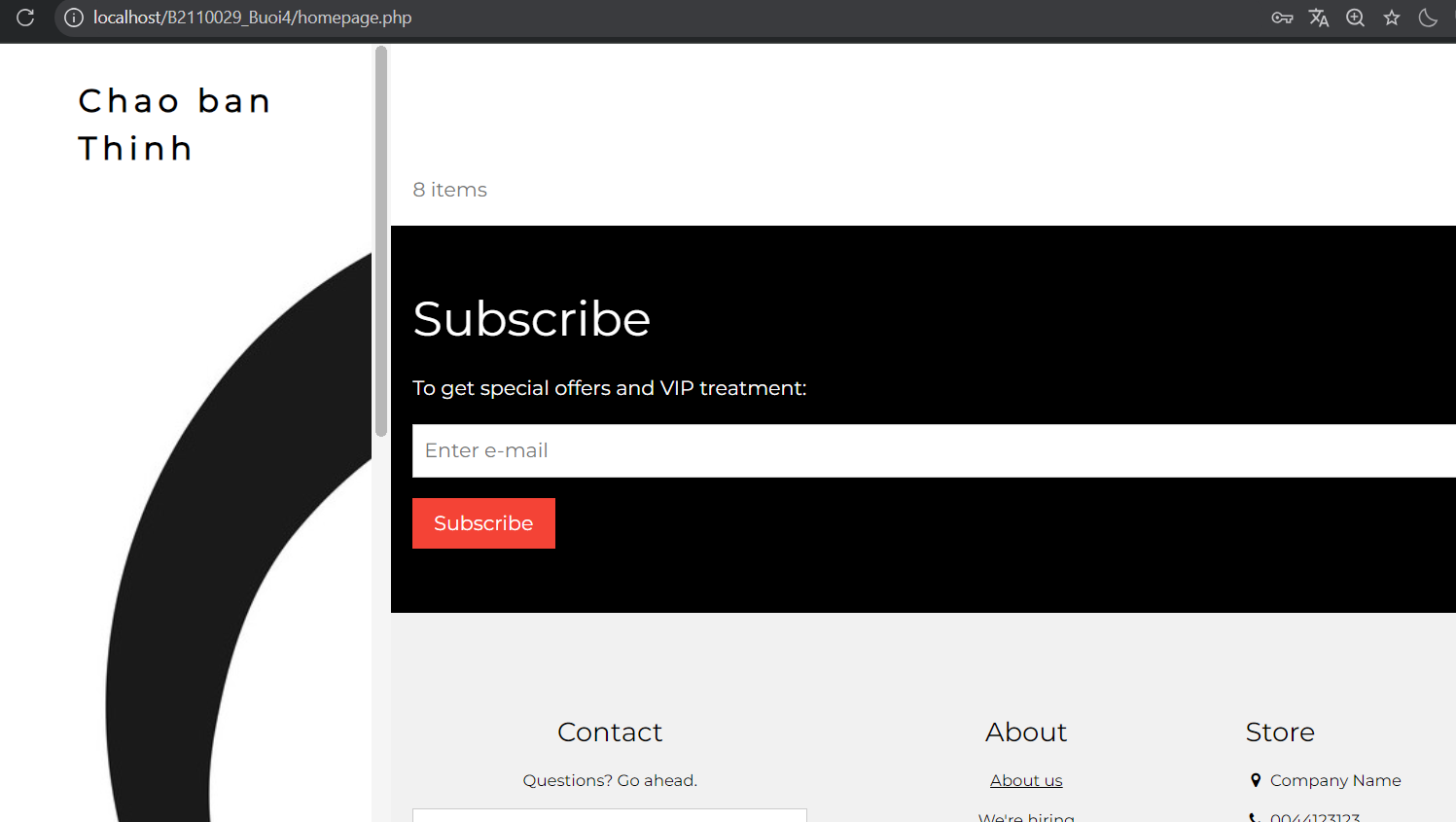




*(check lại CSDL thấy được ngay mật khẩu đã được băm thành mã khác🡪mật khẩu đã được thay đổi)*

- Thử đăng nhập lại bằng mật khẩu mới (mật khẩu mới của mình là 123456, mật khẩu cũ vừa bị đổi là 12345)





*(3 câu 8, 9, 10 là 3 câu đọc thêm, các bạn có thể đọc tham khảo)*

**8.**

**9.**

**10.**

# Chú ý:

* Các bạn nộp file word: Đặt tên file: <mssv><hoten><bai4>.docx nộp lên classroom, kèm với các file code đã tạo (**ngoại trừ file word trả lời câu hỏi, các file còn lại các bạn nén vào file .zip**).
* Mỗi câu các bạn trả lời bằng hình/trả lời dạng text tùy vào yêu cầu của câu hỏi, và ghi rõ số thứ tự câu hỏi mà bạn trả lời.
* Các bài phát hiện copy từ các bài của bạn khác sẽ không được tính điểm.
* *Chú ý: Các câu trả lời có tham khảo trên Internet phải trích dẫn link*