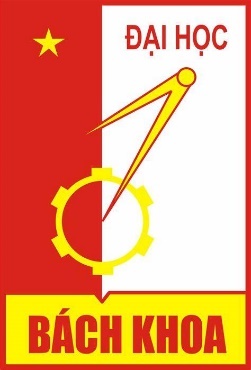
TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI

**VIỆN ĐÀO TẠO LIÊN TỤC**

-----🙞🙜🕮🙞🙜-----

**BÁO CÁO PROJECT 2**

**Tìm hiểu, áp dụng Google Authenticator xây dựng website có cơ chế xác thực hai yếu tố**

|  |  |
| --- | --- |
| *Giáo viên hướng dẫn:* | **ThS. Bùi Trọng Tùng** |
| *Sinh viên thực hiện:* | **Phạm Thành Vân** |
|  |  |

**Hà Nội 12/2020**

MỤC LỤC

[DANH MỤC HÌNH ẢNH 2](#_Toc60015166)

[DANH MỤC BẢNG 3](#_Toc60015167)

[LỜI MỞ ĐẦU 4](#_Toc60015168)

[CHƯƠNG 1. Phân tích bài toán 6](#_Toc60015169)

[1.1. Hệ thống xác thực bằng mật khẩu 6](#_Toc60015170)

[1.2. Đe dọa đối với phương thức xác thực bằng mật khẩu 6](#_Toc60015171)

[1.3. Xác thực đa yếu tố 8](#_Toc60015172)

[CHƯƠNG 2. Thiết kế giải pháp 11](#_Toc60015173)

[2.1. Mô hình chức năng 11](#_Toc60015174)

[2.2. Mô hình cấu trúc 16](#_Toc60015175)

[CHƯƠNG 3. Thử nghiệm, đánh giá 19](#_Toc60015176)

[3.1. Mô hình kiến trúc tổng thể hệ thống 19](#_Toc60015177)

[3.2. Kết quả thử nghiệm 20](#_Toc60015178)

[3.2.1. Đăng ký 20](#_Toc60015179)

[3.2.2. Cập nhật thông tin cá nhân 21](#_Toc60015180)

[3.2.3. Đăng xuất 22](#_Toc60015181)

[3.2.4. Đăng nhập thành công 23](#_Toc60015182)

[3.2.4. Đăng nhập không thành công 25](#_Toc60015183)

[KẾT LUẬN 27](#_Toc60015184)

# DANH MỤC HÌNH ẢNH

[Hình 1. Giao diện ứng dụng Google Authenticator trên điện thoại 10](#_Toc60015185)

[Hình 2. Biểu đồ ca sử dụng chức năng của người dùng với hệ thống. 11](#_Toc60015186)

[Hình 3. Biểu đồ hoạt động cho ca sử dụng Đăng ký 12](#_Toc60015187)

[Hình 4. Biểu đồ hoạt động cho ca sử dụng Sửa thông tin 13](#_Toc60015188)

[Hình 5. Biểu đồ ca hoạt động cho ca sử dụng Đăng nhập 15](#_Toc60015189)

[Hình 6. Biểu đồ hoạt động cho ca sử dụng Đăng xuất 16](#_Toc60015190)

[Hình 7. Biểu đồ lớp 17](#_Toc60015191)

[Hình 8. Mô hình kiến trúc tổng thể hệ thống 19](#_Toc60015192)

[Hình 9. Giao diện màn hình đăng ký 20](#_Toc60015193)

[Hình 10. Màn hình thông tin cá nhân của người dùng 21](#_Toc60015194)

[Hình 11. Kết quả khi quét QR code bằng Google Authenticator 22](#_Toc60015195)

[Hình 12. Nút đăng xuất khỏi hệ thống 22](#_Toc60015196)

[Hình 13. Màn hình đăng nhập 23](#_Toc60015197)

[Hình 14. Bước 1 và bước 2 24](#_Toc60015198)

[Hình 15. Bước 3 và bước 4 24](#_Toc60015199)

[Hình 16. Bước 5 và bước 6 25](#_Toc60015200)

[Hình 17. Đăng nhập sai username hoặc password 26](#_Toc60015201)

[Hình 18. Đăng nhập sai OTP 26](#_Toc60015202)

# DANH MỤC BẢNG

[Bảng 1. Mô tả ca sử dụng Đăng ký 12](#_Toc60015203)

[Bảng 2. Mô tả ca sử dụng Sửa thông tin 13](#_Toc60015204)

[Bảng 3. Mô tả ca sử dụng Đăng nhập 14](#_Toc60015205)

[Bảng 4. Mô tả ca sử dụng Đăng xuất 16](#_Toc60015206)

[Bảng 5. Thẻ CRC của lớp User 17](#_Toc60015207)

[Bảng 6. Bảng dữ liệu User 18](#_Toc60015208)

# LỜI MỞ ĐẦU

Hiện nay, khi internet đã trở nên phổ biến, nhu cầu sử dụng các dịch vụ, trao đổi thông tin, dữ liệu của người dùng qua mạng đang tăng lên ngày một mạnh mẽ. Bên cạnh những lợi ích mà internet mang lại cũng tiềm ẩn những hiểm họa và nguy cơ mất an toàn như tài khoản cá nhân có thể bị đánh cắp và sử dụng mào mục đích bất hợp pháp. Chính vì vậy, có rất nhiều giải pháp nhằm đảm bảo an toàn cho thông tin của người sử dụng được triển khai và thiết lập như quản lý tài khoản người dùng thông qua username và password, xác thực khi truy cập và sử dụng dịch vụ, một số giải pháp bảo vệ hệ thống như sử dụng hệ thống pháp hiện xâm nhập, tường lửa để ngăn chặn và cảnh báo truy cập trái phép vào hệ thống, các công cụ quét và phát hiện mã độc hại, v.v. trong đó thì phương pháp xác thực danh tính người dung khi sử dụng một dịch vụ trên mạng là một điều quan trọng, giup đảm bỏa an toàn cho hệ thống cũng như đảm bảo thông tin riêng cho người dùng.

Xác thực danh tính là tạo ra liên kết giữa định danh và đối tượng, thực thể. Xác thực danh tính được thực hiện qua 2 bước:

* Chủ thể cung cấp một định danh trong hệ thống
* Chủ thể cung cấp thông tin xác thực có thể chứng minh sự liên kết giữa định danh và chủ thể

Các phương pháp xác thực chính:

* Cái chủ thể biết
* Cái chủ thể có
* Chủ thể là gì
* Vị trí của chủ thể

Đối với phương pháp xác thực phổ biến là sử dụng tài khoản đăng nhập của người dùng gồm username và password đã bộ lộ những điểm yếu khi có thể bị đánh cắp trong trường hợp máy tính của người dùng bị cài đặt các phần mềm, chương trình có khả năng đánh cắp dữ liệu, hay trong quá trình trao đổi thông tin qua mạng bị nghe lén trên đường truyền, bị chuyển hướng đến trang web chứa mã độc hại, v.v. Để đảm bảo an toàn hơn cho người dùng, các nhà cung cấp dịch vụ đã áp dụng những phương pháp xác thực mạnh hơn khi kết hợp các yếu tố có được từ người dùng như username, password, mã PIN, đặc điểm sinh trắc học, v.v. Trong đó, phương pháp xác thực hai nhân tố với mật khẩu sử dụng một lần (OTP) kết hợp thông tin về username, password của người dùng hiện đang được áp dụng phổ biến.

Project này sẽ tìm hiểu, áp dụng Google Authenticator để xây dựng một website có cơ chế xác thực hai yếu tố với username, password và OTP khi người dùng đăng nhập vào hệ thống.

Nội dung của project bao gồm:

* Chương 1: Phân tích bài toán.

Phân tích các mối đe dọa với phương pháp xác thực bằng username, password truyền thống. Trình bày về xác thực hai yếu tố.

* Chương 2: Thiết kế giải pháp.

Trình bày use cases, mô tả hoạt động của việc xác thực hai yếu tố trong hệ thống, biểu đồ lớp, thiết kế cơ sở dữ liệu.

* Chương 3: Thử nghiệm, đánh giá.

Mô tả hoạt động của phần mềm thử nghiệm xác thực hai yếu tố.

# CHƯƠNG 1. Phân tích bài toán

## Hệ thống xác thực bằng mật khẩu

Các thành phần của hệ xác thực bao gồm:

**A**: Tập các thông tin đặc trưng mà chủ thể sử dụng để chứng minh định danh của mình.

**C**: Tập các thông tin mà hệ thống lưu trữ và sử dụng để xác minh sự đúng đắn của thông tin trong tập **A**.

**F**: Tập các hàm sinh **C** từ **A**

**L**: Tập các hàm xác thực

**S**: Tập các hàm lựa chọn cho phép các thực thể tạo hoặc thay thế các thông tin trong **A** và **C**.

Hệ xác thực mật khẩu, giả sử mật khẩu lưu dưới dạng rõ:

**A**: tập các chuỗi ký tự được chấp nhận là mật khẩu.

**C = A**

**F**: hàm đồng nhất thức I

**L**: hàm so sánh =

**S**: hàm thiết lập, thay đổi mật khẩu.

## Đe dọa đối với phương thức xác thực bằng mật khẩu

Một số điểm yếu trên hệ thống xác thực bằng mật khẩu:

* Lưu trữ mật khẩu trong CSDL không an toàn.
* Truyền mật khẩu trên kênh không an toàn
* Người dùng không cẩn trọng:
  + Sử dụng mật khẩu yếu
  + Ghi chép mật khẩu vào văn bản
  + Chi sẻ mật khẩu cho người khác (vô tình hoặc cố ý)
  + v.v.

Thông thường, chúng ta thường đổ lỗi cho người dùng khi họ sơ ý bị kẻ tấn công khai thác. Tuy nhiên, chúng ta cần xây dựng hệ thống có khả năng hỗ trợ người dùng không hành động sai.

* **Tấn công vào hệ xác thực bằng mật khẩu**
* Tấn công thụ động: nghe lén, quan sát quá trình nhập mật khẩu
  + Nhìn trộm
  + Sử dụng chương trình key logging
  + Tấn công kênh bên
  + Chặn bắt gói tin
* Tấn công chủ động:
  + Giả mạo chương trình cung cấp dịch vụ (server)
  + Giả mạo chương trình khách (client)
  + Tấn công man-in-the-middle
  + Tấn công vào máy chủ vật lý cung cấp dịch vụ
* **Tấn công dạng online**
* Kẻ tấn công biết tập hàm xác thực **L**
* Mục đích: dò thử lần lượt các mật khẩu dựa trên kết quả xác thực hệ thống trả lại
* Đặc điểm:
  + Tương tác trực tiếp với hệ xác thực
  + Có thể thử trên 1 hoặc đồng thời nhiều tài khoản
* Xác suất tấn công thành công:
  + G: tốc độ kẻ tấn công dò thử
  + T: thời gian kẻ tấn công dò thử
  + N: số mật khẩu hệ thống có thể tạo ra
* Giảm thiểu:
  + Tăng độ dài của mật khẩu
  + Quy định số lần thử xác thực tối đa trong một khoảng thời gian
* **Tấn công dạng off-line**
* Kẻ tấn công biết:
  + Tập thông tin **C** hệ thống dùng để xác thực
  + Các hàm biến đổi **F**
* Mục tiêu: tìm các thông tin
* Đặc điểm: không tương tác với hệ xác thực
* Ví dụ: kẻ tấn công biết có CSDL chứa mã băm của mật khẩu vả hàm băm sử dụng.
* Nguy cơ: người dùng sử dụng các mật khẩu dễ đoán, kẻ tấn công có một bộ từ điển chứa mã băm tương ứng
* Giảm thiểu nguy cơ: băm mật khẩu với “salt”
* **Một số chính sách sử dụng mật khẩu**

Mục đích: tăng cường an toàn cho hệ xác thực dựa trên mật khẩu

* Quy định độ dài tối thiểu
* Quy định các ký tự bắt buộc phải sử dụng
* Thay đổi mật khẩu định kỳ
* Hạn chế sử dụng lại mật khẩu cũ trong một khoảng thời gian nhất định
* Hạn chế số lần thử xác thực
* Tăng thời gian chờ thử xác thực lại
* Yêu cầu đổi mật khẩu sau lần đăng nhập đầu tiên

## Xác thực đa yếu tố

Phương pháp xác thực sử dụng mật khẩu không đủ an toàn, nguyên nhân chủ yếu từ người dùng. Để sử dụng mật khẩu một cách an toàn như sử dụng mật khẩu đủ dài và khó đoán, không dùng chung cho nhiều tài khoản, thay đổi thường xuyên, v.v. thì hầu hết người dùng không thực hiện được. Vì vậy, cần thêm các yếu tố xác thực an toàn hơn, không phụ thuộc vào thói quen của người dùng. Mức độ an toàn, bảo mật càng cao khi số yếu tố xác thực càng nhiều. Tuy nhiên, khi số yếu tố xác thực lớn thì hệ thống càng phức tạp, kéo theo chi phí đầu tư và duy trì vận hành tốn kém, đồng thời lại bất tiện cho người sử dụng. Do vậy, trên thực tế để cân bằng giữa an toan, bảo mật và tính tiện dụng, hiện nay sử dụng phổ biến nhất là xác thực 2 yếu tố:

* Cái người dùng biết: mật khẩu
* Cái người dùng có: thường là thiết bị phần cứng.

Ví dụ như xác thực chủ thẻ trong giao dịch ATM, yếu tố xác thực đầu tiên của khách hàng là thẻ ATM (cái khách hang có), sau khi đưa thẻ vào máy, khách hàng sẽ phải đưa tiếp yếu tố xác thực thứ hai là số PIN (cái khách hàng biết). Một ví dụ khác là xác thực người sử dụng dịch vụ giao dịch Internet Banking: khách hàng đăng nhập với username và password sau đó còn phải cung cấp tiếp OTP (One Time Password) được sinh ra trên token riêng của khách hàng.

* **Mật khẩu sử dụng một lần (One Time Password - OTP)**

Mật khẩu sử dụng một lần là loại mật khẩu chỉ dùng một lần và chỉ có giá trị cho một phiên đăng nhập hoặc cho một giao dịch trong một khoảng thời gian nhất định.

Mục đích của mật khẩu OTP là làm cho mật khẩu lúc nào cũng thay đổi, tránh việc kẻ tấn công dễ dàng đánh cắp tài khoản của người dùng và sử dụng vào mục đích bất hợp pháp. Yêu cầu đối với mật khẩu OTP là có thể chống lại được cách thức tấn công phát lại, có nghĩa là trong trường hợp có được thông tin về OTP trong một phiên làm việc thì cũng không thể sử dụng nó trong phiên làm việc kế tiếp, không thể đoán được mật khẩu tiếp theo để sử dụng.

Phân loại:

* S/Key OTP
* Hash-based OTP (HOTP)
* Time-based OTP (TOTP)

Cách thức phân phối:

* SMS
* Ứng dụng
* Email
* Token
* **HOTP**

Cách sinh mật khẩu OTP sử dụng thuật toán HOTP:

HOTP(K, C) = Truncate(HMAC-SHA-1(K, C))

Trong đó:

C: bộ đếm đã được đồng bộ giữa client và server. C có độ dài 8 byte

K: giá trị chi sẻ bí mật giữa client và server.

Truncate(): là hàm chuyển giá trị HMAC-SHA-1 thành giá trị HOTP.

Có thể mô tả quá trình sinh giá trị HTOP thành 3 bước như sau:

* Bước 1: Sinh giá trị HMAC-SHA-1. Đặt HS = HMAC-SHA-1(K, C). HS là chuỗi 20 byte
* Bước 2: Sinh chuỗi 4 byte. Đặt Sbits = DT(HS). Trong đó hàm DT được định nghĩa như sau:
  + Đầu vào: chuỗi 20 byte S (S = S[0]...S[19])
  + Xử lý:
    - Lấy OffsetBits = 4 bit thấp của S[19]
    - Biến đổi sang dạng thập phân: Offset = StToNum(OffsetBits)
    - Trích xuất 4 byte trong chuỗi S bắt đầu từ vị trí Offset được chuỗi P.
    - Bỏ bit đầu tiên của chuỗi P, trả về 31 bit sau của chuỗi P.
* Bước 3: Tính giá trị HOTP

Đặt Snum = StToNum(Sbits) (chuyển S bit sang dạng thập phân)

Trả về D = Snum mod 10d (d là số chữ số của OTP – thông thường có độ dài từ 6 đến 8 chữ số)

Ví dụ sinh giá trị HOTP với 6 chữ số:

Giả sử HS = HMAC-SHA-1(K,C) có giá trị như sau:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Thứ tự | | | | | | | | | | | | | | | | | | | |
| 00 | 01 | 02 | 03 | 04 | 05 | 06 | 07 | 08 | 09 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 |
| Giá trị | | | | | | | | | | | | | | | | | | | |
| 1f | 86 | 98 | 69 | 0e | 02 | ca | 16 | 61 | 85 | 50 | ef | 7f | 19 | da | 8e | 94 | 5b | 55 | 5a |

* Byte cuối cùng (byte 19) có giá trị 0x5a
* Giá trị của 4 bit thấp là 0xa (OffsetBits)
* Giá trị thập phân của 0xa là 10 (Offset)
* Giá trị của 4 byte bắt đầu từ vị trí thứ 10 là 0x50ef7f19 (P)
* Lấy 31 bit sau của P, đổi sang giá trị thập phân và chia cho 106ta thu được kết quả là 872921
* **TOTP**

Thuật toán TOTP thực hiện tương tự HOTP nhưng dựa trên thời gian T thay vì giá trị đếm C.

TOTP = HOTP(K, T)

Trong đó:

T: là số nguyên biểu diễn số bước thời gian từ thời điểm T0 đến thời điểm Unix hiện tại (được tính từ thời điểm Unix Epoch là ngày 01/01/2970 theo UTC).

T = (Current Unix Time – T0)/X

T0: giá trị ban đầu (thường chọn T0 = 0)

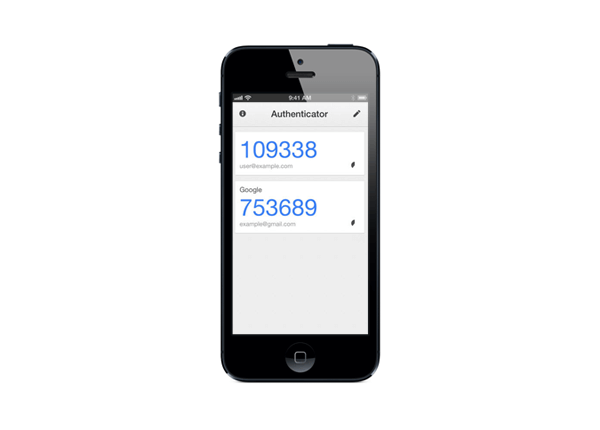
X: bước thời gian. Là tham số quyết định thời gian hợp lệ của mật khẩu

Trong project sẽ sử dụng xác thực 2 yếu tố dựa trên mật khẩu và TOTP sinh bởi ứng dụng Google Authenticator trên điện thoại của người dùng.

* **Google Authenticator**

Google Authenticator là một ứng dụng xác minh 2 yếu tố trên điện thoại giúp bảo mật tốt hơn cho các tài khoản bằng cách yêu cầu bược xác mình thứ 2 khi đăng nhập.

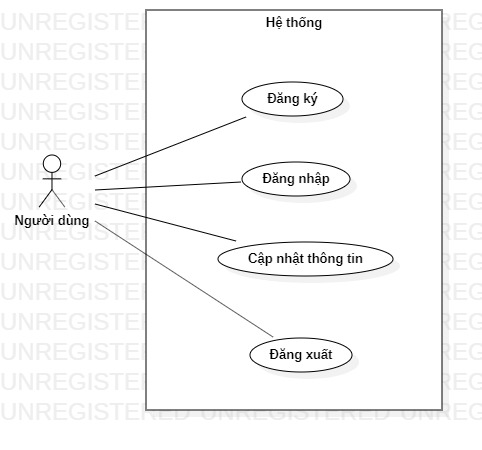
Google Authenticator sinh mã xác thực bao gồm 6 số có thời hạn sử dụng trong vòng 30 giây.



Hình 1. Giao diện ứng dụng Google Authenticator trên điện thoại

# CHƯƠNG 2. Thiết kế giải pháp

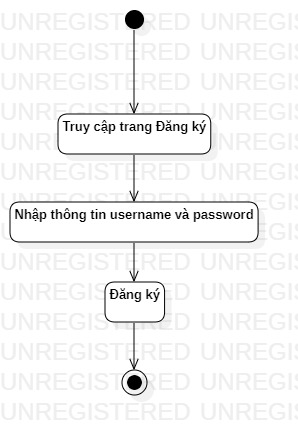
## Mô hình chức năng



Hình 2. Biểu đồ ca sử dụng chức năng của người dùng với hệ thống.

|  |  |  |
| --- | --- | --- |
| Tên ca sử dụng: Đăng ký | ID: 1 | Mức độ quan trọng: Trung bình |
| Tác nhân chính: Người dùng | Loại ca sử dụng: | |
| Các bên liên quan và mối quan tâm: Người dùng đăng ký hệ thống | | |
| Mô tả ngắn gọn: Ca sử dụng này mô tả cách người dùng đăng ký vào hệ thống | | |
| Kích hoạt: Người dùng đăng ký hệ thống  Loại: Ngoại | | |
| Các mối liên hệ:  Tương tác: Người dùng, hệ thống hiện có  Bao gồm:  Mở rộng:  Khái quát hóa: | | |
| Luồng sự kiện thông thường:   1. Người dùng truy cập vào trang đăng ký 2. Hiển thị màn hình đăng ký 3. Người dùng nhập username và password 4. Người dùng bấm nút đăng ký. 5. Hiển thị trang thông tin của người dùng | | |
| Các luồng con: | | |
| Các luồng ngoại lệ/Tương đương: | | |

Bảng 1. Mô tả ca sử dụng Đăng ký



Hình 3. Biểu đồ hoạt động cho ca sử dụng Đăng ký

|  |  |  |
| --- | --- | --- |
| Tên ca sử dụng: Sửa thông tin | ID: 2 | Mức độ quan trọng: Trung bình |
| Tác nhân chính: Người dùng | Loại ca sử dụng: | |
| Các bên liên quan và mối quan tâm: Người dùng sửa thông tin cá nhân | | |
| Mô tả ngắn gọn: Ca sử dụng này mô tả cách người dùng sửa thông tin cá nhân | | |
| Kích hoạt: Người dùng sửa thông tin cá nhân  Loại: Ngoại | | |
| Các mối liên hệ:  Tương tác: Người dùng, hệ thống hiện có  Bao gồm:  Mở rộng:  Khái quát hóa: | | |
| Luồng sự kiện thông thường:   1. Đã truy cập thành công vào hệ thống 2. Hiển thị màn hình thông tin cá nhân của người dùng 3. Người dùng sửa các thông tin cá nhân nếu cần thiết 4. Người dùng lưu các thông tin đã được sửa. | | |
| Các luồng con: | | |
| Các luồng ngoại lệ/Tương đương: | | |

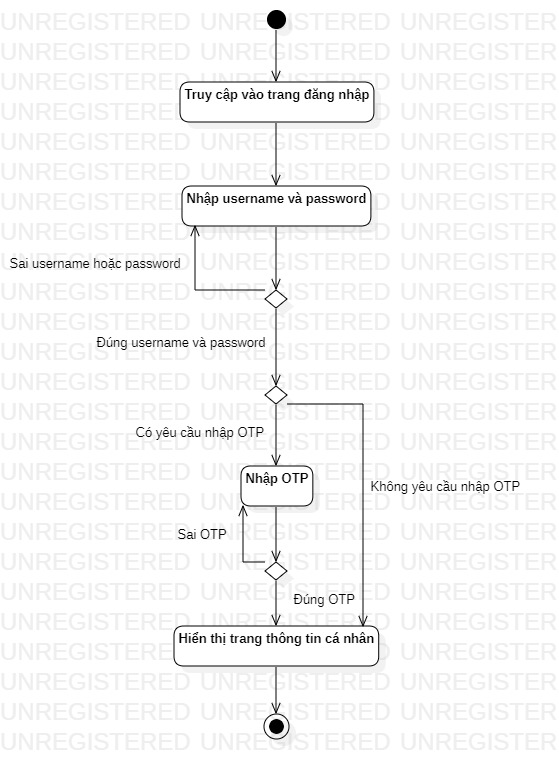
Bảng 2. Mô tả ca sử dụng Sửa thông tin



Hình 4. Biểu đồ hoạt động cho ca sử dụng Sửa thông tin

|  |  |  |
| --- | --- | --- |
| Tên ca sử dụng: Đăng nhập | ID: 3 | Mức độ quan trọng: Cao |
| Tác nhân chính: Người dùng | Loại ca sử dụng: | |
| Các bên liên quan và mối quan tâm: Người dùng đăng nhập vào hệ thống | | |
| Mô tả ngắn gọn: Ca sử dụng này mô tả cách người dùng đăng nhập vào hệ thống | | |
| Kích hoạt: Người dùng đăng nhập vào hệ thống  Loại: Ngoại | | |
| Các mối liên hệ:  Tương tác: Người dùng, hệ thống hiện có  Bao gồm:  Mở rộng:  Khái quát hóa: | | |
| Luồng sự kiện thông thường:   1. Người dùng mở trang đăng nhập 2. Hiển thị màn hình đăng nhập 3. Người dùng nhập thông tin username và password 4. Nếu thông tin username và password chính xác thì người dùng người dùng đăng nhập thành công và hiển thị trang thông tin người dùng | | |
| Các luồng con: | | |
| Các luồng ngoại lệ/Tương đương:  4a. Nếu sai thông tin username hay password thì hiển thị thông báo và người dùng vẫn ở trang đăng nhập  4b. Nếu thông tin username và password chính xác nhưng có yêu cầu xác thực 2 nhân tố thì hiển thị trang nhập OTP   1. Người dùng nhập OTP 2. Nếu thông tin OTP chính xác thì người dùng đăng nhập thành công và hiển thị trang thông tin người dùng   6a. Nếu thông tin OTP không chính xác thì hiển thị thông báo | | |

Bảng 3. Mô tả ca sử dụng Đăng nhập



Hình 5. Biểu đồ ca hoạt động cho ca sử dụng Đăng nhập

|  |  |  |
| --- | --- | --- |
| Tên ca sử dụng: Đăng xuất | ID: 4 | Mức độ quan trọng: Trung bình |
| Tác nhân chính: Người dùng | Loại ca sử dụng: | |
| Các bên liên quan và mối quan tâm: Người dùng đăng xuất khỏi hệ thống | | |
| Mô tả ngắn gọn: Ca sử dụng này mô tả cách người dùng đăng xuất khỏi hệ thống | | |
| Kích hoạt: Người dùng đăng xuất khỏi hệ thống  Loại: Ngoại | | |
| Các mối liên hệ:  Tương tác: Người dùng, hệ thống hiện có  Bao gồm:  Mở rộng:  Khái quát hóa: | | |
| Luồng sự kiện thông thường:   1. Đã truy cập thành công vào hệ thống 2. Hiển thị màn hình thông tin cá nhân của người dùng 3. Người dùng thực hiện đăng xuất 4. Hiển thị màn hình đăng nhập | | |
| Các luồng con: | | |
| Các luồng ngoại lệ/Tương đương: | | |

Bảng 4. Mô tả ca sử dụng Đăng xuất

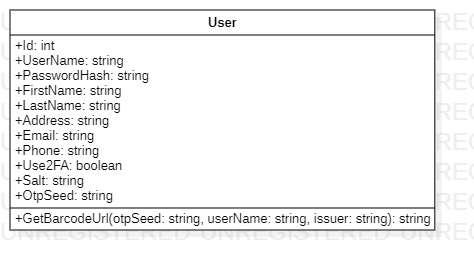


Hình 6. Biểu đồ hoạt động cho ca sử dụng Đăng xuất

## Mô hình cấu trúc

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Mặt trước**   |  |  |  | | --- | --- | --- | | **Tên lớp**: User | **ID**: 1 | **Kiểu**: Cụ thể | | **Mô tả**: Người dùng trong hệ thống | | **Ca sử dụng liên quan**: 1, 2, 3, 4 | | **Các trách nhiệm** | | **Các đối tác** |   **Mặt sau**   |  | | --- | | **Các thuộc tính:**  Id  UserName  PasswordHash  FirstName  LastName  Address  Email  Phone  Use2FA  Salt  OtpSeed | | **Các mối quan hệ:**  **Khái quát hoá (thuộc loại):**  **Tổng hợp (Có các phần):**  **Các mối liên quan khác:** | |

Bảng 5. Thẻ CRC của lớp User



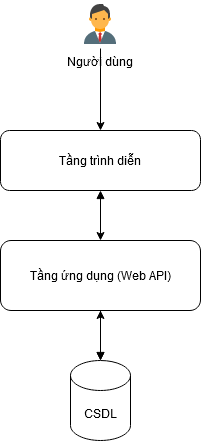
Hình 7. Biểu đồ lớp

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên cột** | **Kiểu dữ liệu** | **Được phép NULL** | **Ghi chú** |
| 1 | Id | int | Không | Khoá chính |
| 2 | UserName | nvarchar(50) | Không |  |
| 3 | PasswordHash | nvarchar(50) | Không |  |
| 4 | FirstName | nvarchar(50) | Có |  |
| 5 | LastName | nvarchar(50) | Có |  |
| 6 | Address | nvarchar(500) | Có |  |
| 7 | Email | nvarchar(50) | Có |  |
| 8 | Phone | nchar(25) | Có |  |
| 9 | Use2FA | bit | Không |  |
| 10 | Salt | nvarchar(50) | Có |  |
| 11 | OtpSeed | nvarchar(50) | Có |  |

Bảng 6. Bảng dữ liệu User

# CHƯƠNG 3. Thử nghiệm, đánh giá

## 3.1. Mô hình kiến trúc tổng thể hệ thống

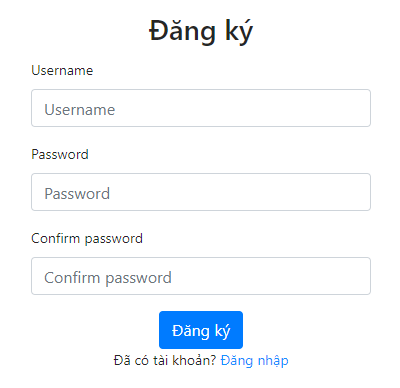


Hình 8. Mô hình kiến trúc tổng thể hệ thống

* Tầng trình diễn: Giao diện ứng dụng tương tác trực tiếp với người dùng. Được xây dựng sử dụng ReactJS.
* Tầng ứng dụng: Cung cấp Web API xử lý yêu cầu từ tầng trình diễn và phản hồi lại cho tầng trình diễn. Xây dựng sử dụng Spring Boot.
* Cơ sở dữ liệu: Lưu dữ liệu của hệ thống. Sử dụng MS SQL Server 2019 Express

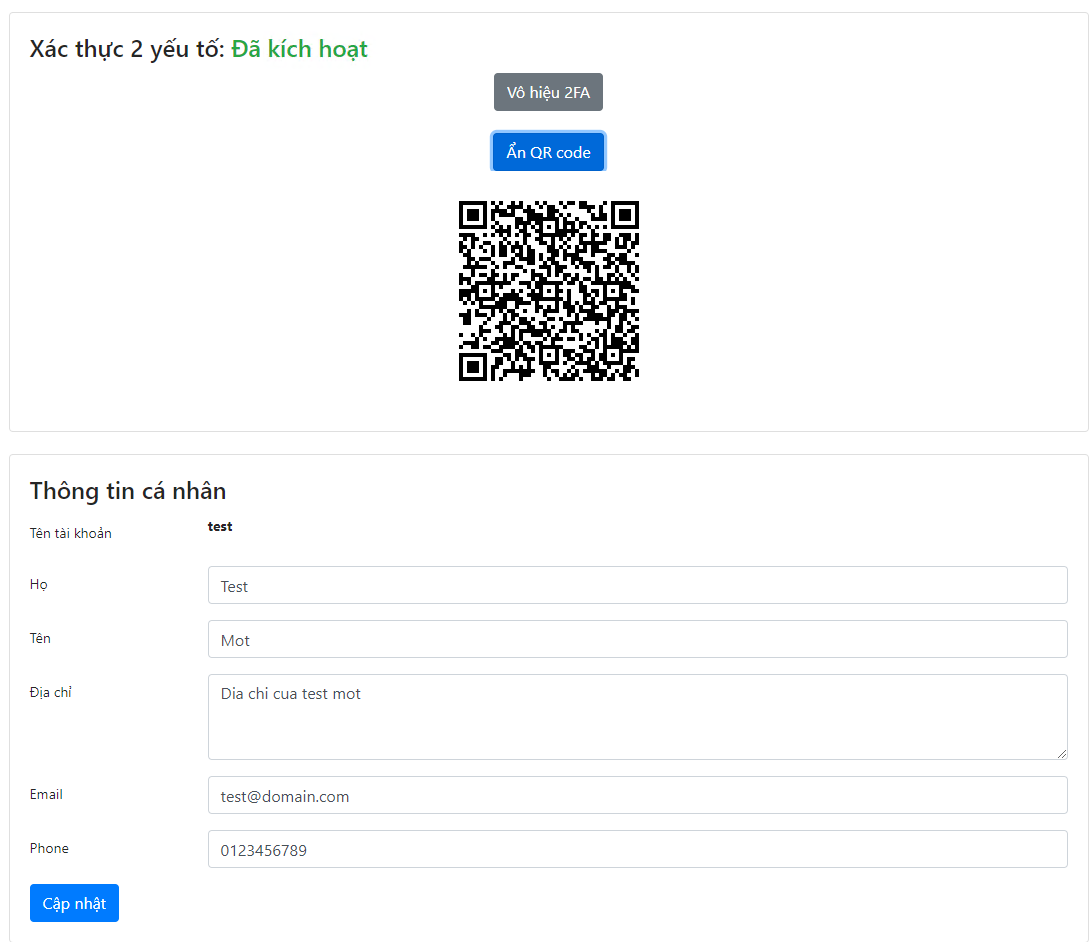
## 3.2. Kết quả thử nghiệm

### 3.2.1. Đăng ký



Hình 9. Giao diện màn hình đăng ký

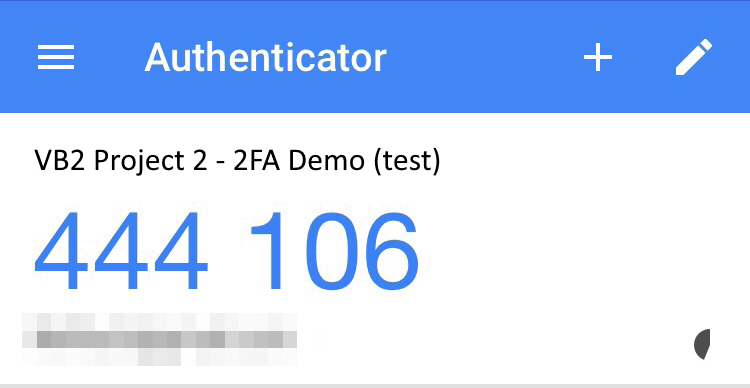
* Người dùng truy cập vào trang đăng ký
* Nhập các trường **Username, Password** và **Confirm Password** (phải trùng giá trị với trường **Password**)
* Bấm nút **Đăng ký**
* Khi đăng ký thành công sẽ chuyển sang màn hình thông tin của người dùng



Hình 10. Màn hình thông tin cá nhân của người dùng

### 3.2.2. Cập nhật thông tin cá nhân

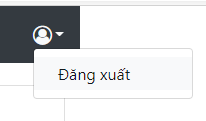
* Kích hoạt/Vô hiệu tính năng xác thực 2 yếu tố bắng các bấm nút: **Kích hoạt 2FA** (**Vô hiệu 2FA**)
  + Khi tính năng xác thực 2 yếu tố được kích hoạt thì có thể Ẩn/Hiện mã QR code để sử dụng Google Authenticator trên điện thoại.
  + Khi quét QR code bằng ứng dụng Google Authenticator sẽ có kết quả tương tự như sau:



Hình 11. Kết quả khi quét QR code bằng Google Authenticator

* Cập nhận thông tin cá nhân bằng cách cập nhật các trường:
  + Họ
  + Tên
  + Địa chỉ
  + Email
  + Số điện thoại
* Sau đó bấm nút **Cập nhật**

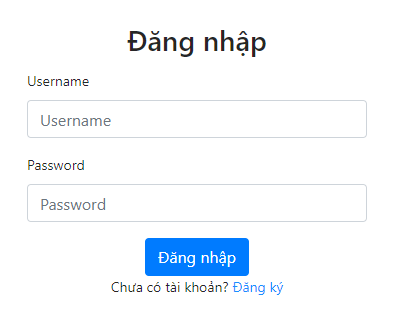
### 3.2.3. Đăng xuất



Hình 12. Nút đăng xuất khỏi hệ thống

* Người dùng thực hiện đăng xuất khỏi hệ thống bằng cách bấm vào biểu tượng người dùng trên thanh Navigation Bar và bấm nút **Đăng xuất**
* Khi đăng xuất thành công sẽ hiển thị màn hình đăng nhập (Hình 12)

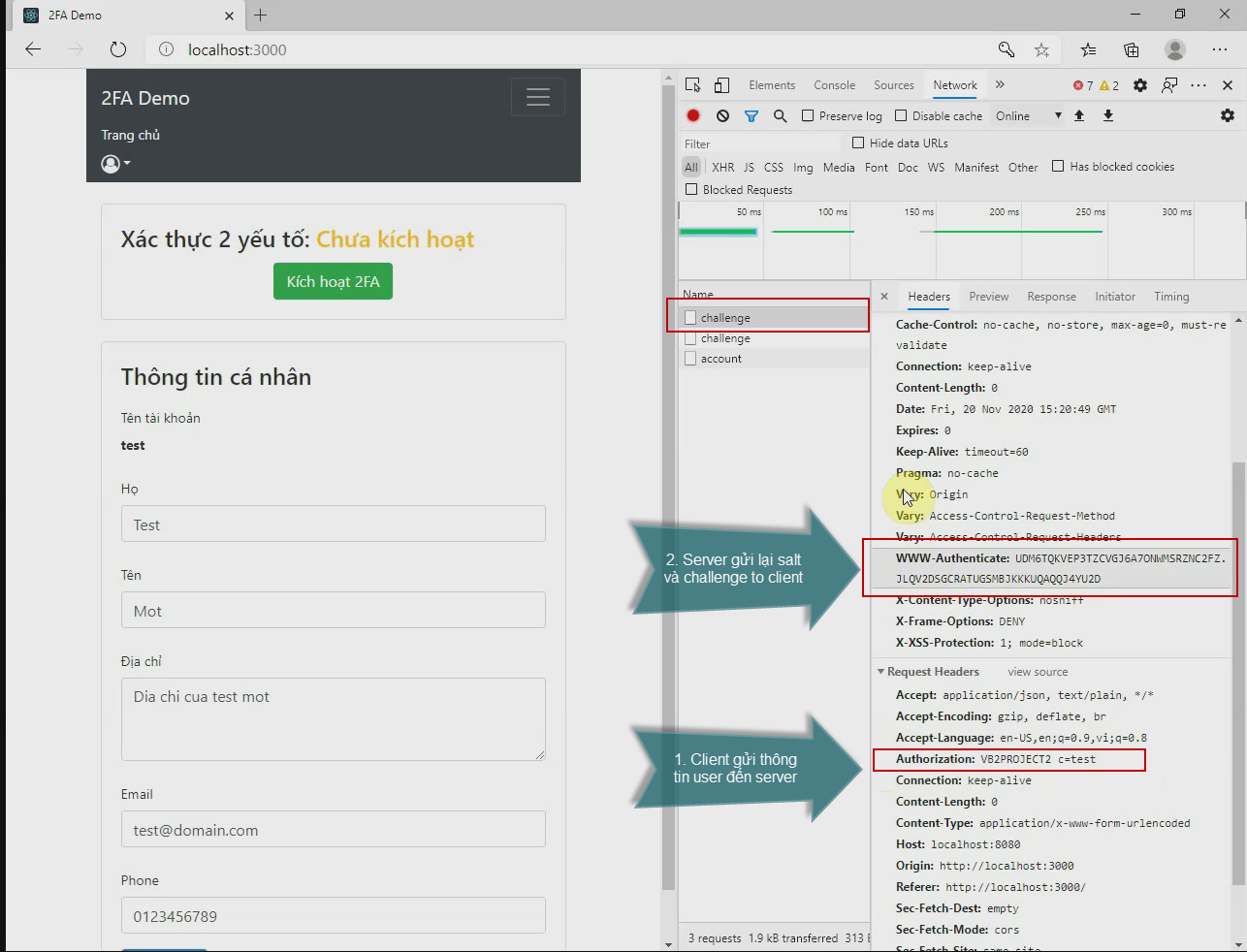
### 3.2.4. Đăng nhập thành công



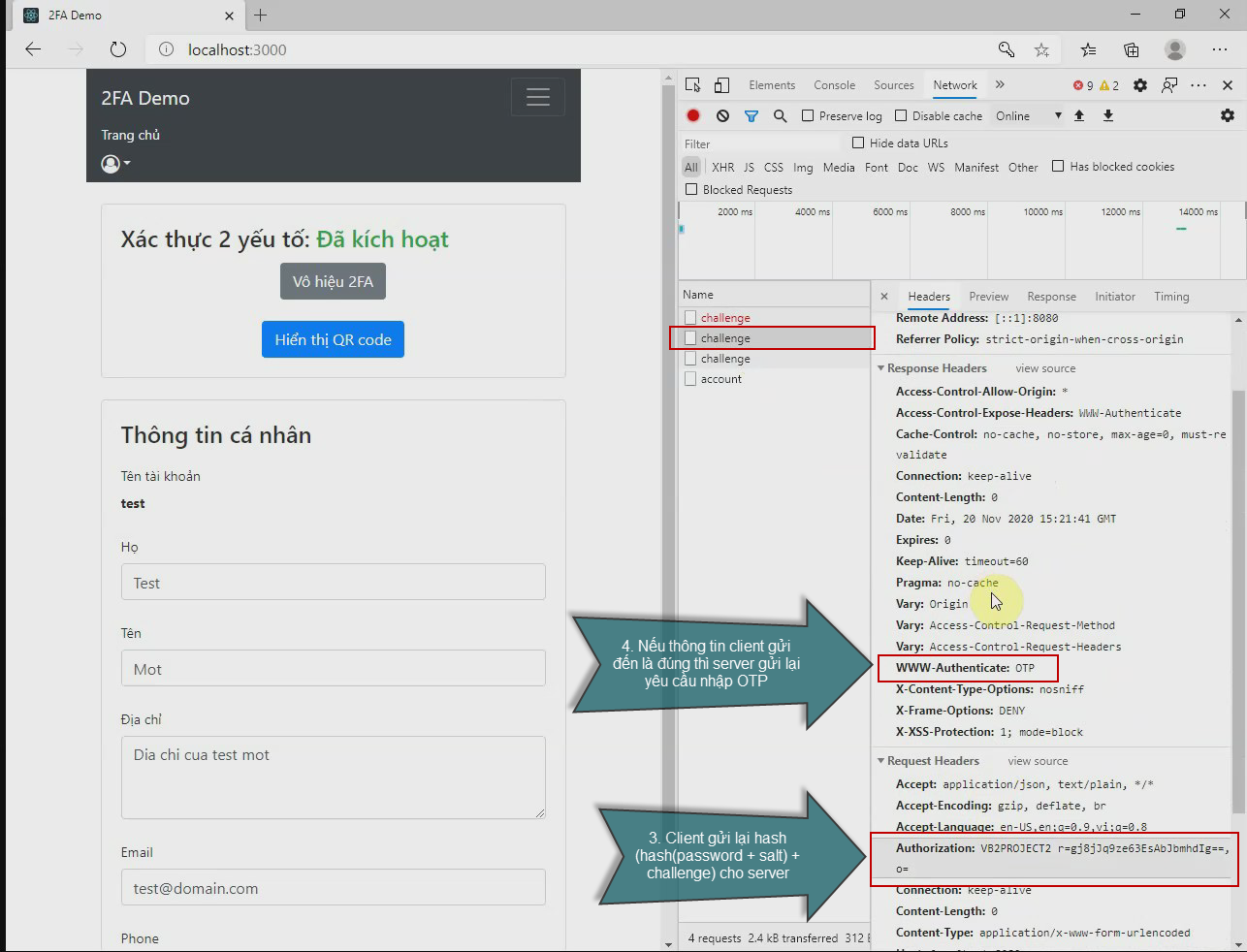
Hình 13. Màn hình đăng nhập

Khi người dùng đăng nhập có sử dụng xác thực 2 yếu tố thì sẽ qua các bước xử lý sau:

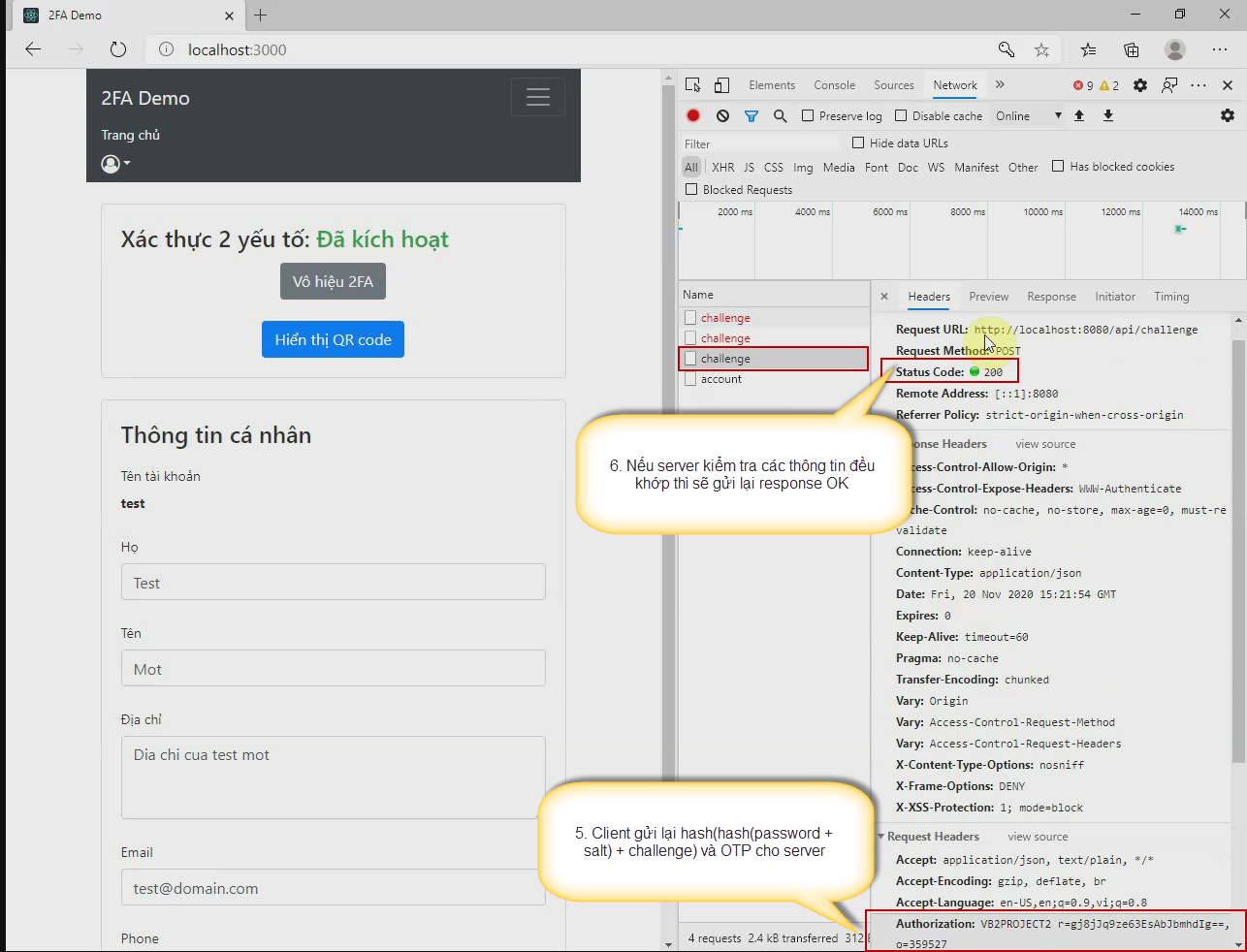
* Bước 1: Client gửi thông tin username cho server.
* Bước 2: Server gửi lại “salt” và challenge cho client. Nếu username sai thì server cũng trả lại “salt” và challenge cho client là chuỗi ngẫu nhiên, không lấy từ CSDL.
* Bước 3: Client gửi lại hash(hash(password+salt) + challenge) cho server.
* Bước 4: Nếu thông tin client gửi đến là đúng thì server gửi yêu cầu nhập OTP
* Bước 5: Client gửi lại thông tin hash(hash(password+salt) + challenge) và OTP cho server.
* Bước 6: Nếu thông tin client gửi lại là đúng thì server trả lại response OK.



Hình 14. Bước 1 và bước 2



Hình 15. Bước 3 và bước 4

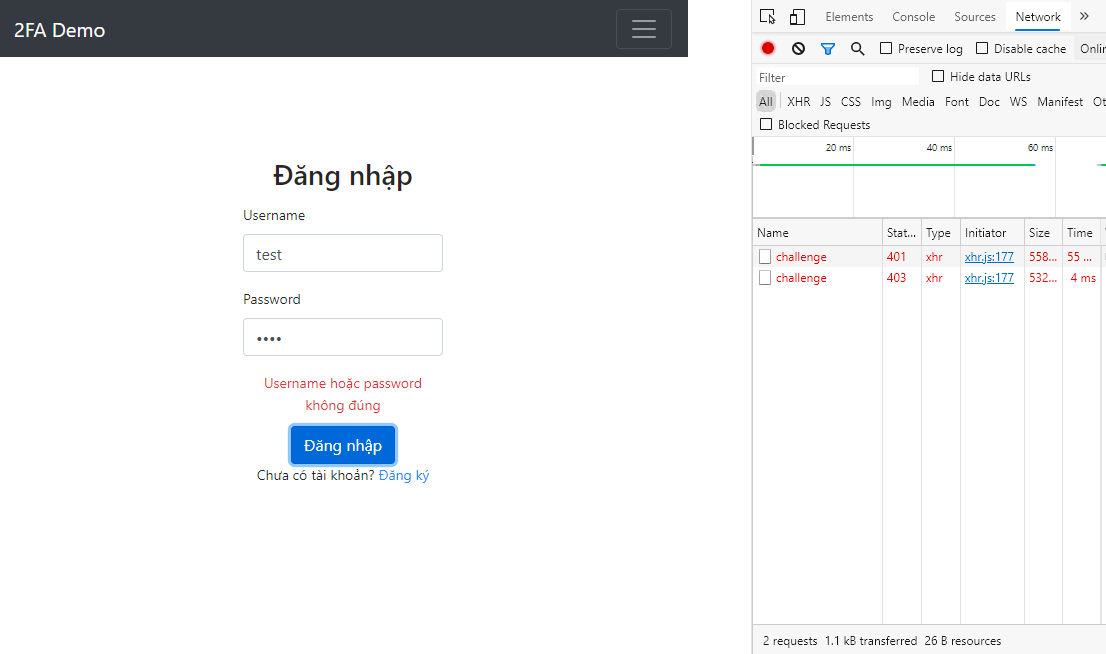


Hình 16. Bước 5 và bước 6

### 3.2.4. Đăng nhập không thành công

* **Sai username hoặc password**

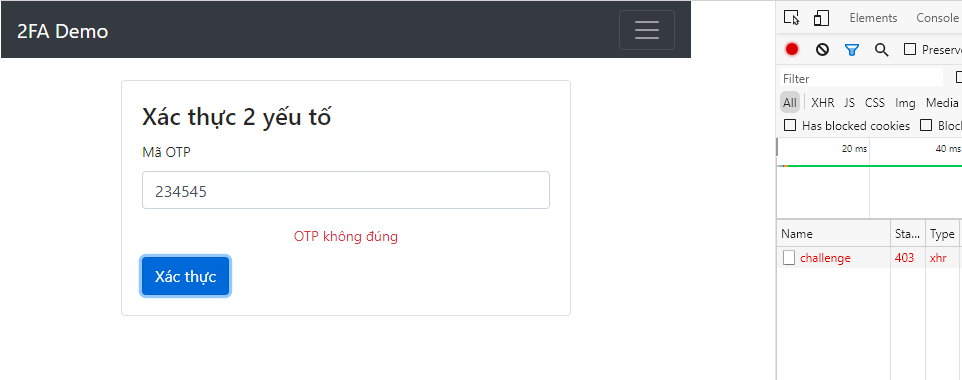
Nếu người dùng nhập sai username hoặc password khi đăng nhập thì server sẽ trả về mã 403 và trên giao diện hiển thị thông báo “**Username hoặc password không đúng**”



Hình 17. Đăng nhập sai username hoặc password

* **Sai OTP**

Nếu người dùng nhập sai OTP thì server trả về mã 403 và trên giao diện hiển thị thông báo OTP không đúng



Hình 18. Đăng nhập sai OTP

# KẾT LUẬN

Phương pháp xác thực 2 nhân tố sử dụng Google Authenticator mang lại sự an toàn thông tin và tiện dụng cho người sử dụng, và hiện phương pháp này cùng đang được sử dụng rộng rãi trong nhiều lĩnh vực và các ứng dụng khác nhau.

Trong project đã sử dụng băm mật khẩu với “salt” sử dụng thuật toán SHA-1 khi lưu vào CSDL giúp bảo mật mật khẩu. Hơn nữa, trong phần xác thực username và password cũng áp dụng giao thức CHAP (Challenge Handshake Authentication Protocol) để đảm bảo an toàn bản thân mật khẩu không bao giờ được gửi qua mạng, giúp bảo mật ngay cả khi không triển khai bảo mật đường truyền.

Các triển khai trong project có thể đưa vào áp dụng trong các các ứng dụng khác trong khâu đăng nhập để đảm bảo tính bảo mật cho thông tin của người dùng.

Do thời gian có hạn nên trong project chưa triển khai được việc xử lý việc mất đồng bộ về thời gian khi sinh mật khẩu OTP dẫn đến mật khẩu sinh OTP trên máy chủ và máy trạm không giống nhau. Project cũng đã thực hiện xác thực bằng token sau khi người dùng đăng nhập thành công để thực hiện các hành động khác trong hệ thông nhưng chưa triển khai được việc bảo mật cho token. Ứng dụng cũng chưa hạn chế số lần đăng nhập sai của người dùng. Những thiếu sót này sẽ được hoàn thiện trong tương lai.