|  |  |  |
| --- | --- | --- |
|  | TRƯỜNG CAO ĐẲNG CÔNG NGHỆ THỦ ĐỨC  **Khoa Công Nghệ Thông Tin** | Macintosh HD:Users:hoangnguyen:Documents:TDC:logoTDC_blue.png |
|  |  |  |

**CHUYÊN ĐỀ WEB 2** | HKI – 2021

PRJ – REPORT

**BÁO CÁO QUÁ TRÌNH MÔN HỌC**

**NHÓM: D**

**THÀNH VIÊN:**

Trần Trọng Hiếu - 19211TT1201

Ngô Bảy Hên – 19211TT1120

Nguyễn Huỳnh Quốc Trung - 19211TT0744

**Mã lớp học phần:** 21111TNC10711102

Table of Contents

[1. PHP Unit Test 3](#_Toc88388313)

[2.1. Function Sumb. 4](#_Toc88388314)

[2.2. Function Auth. 6](#_Toc88388315)

[2.3. Function getUsers. 8](#_Toc88388316)

[2.4. Function insertUser. 9](#_Toc88388317)

[2.4. Function updateUser. 11](#_Toc88388318)

[2. DỰ ÁN CHUYÊN ĐỀ WEB 2 13](#_Toc88388319)

[3.1. Mô Tả Công Nghệ Được Sử Dụng 13](#_Toc88388320)

[3.1.1. ReactJS là gì? 13](#_Toc88388321)

[3.1.2. Redux là gì? 13](#_Toc88388322)

[3.1.3. NextJS là gì? 14](#_Toc88388323)

[3.2. Đặc Tả Yêu Cầu Dự Án 15](#_Toc88388324)

[3.2.1. Chức Năng Đăng Nhập, Đăng Ký, Đăng Xuất 16](#_Toc88388325)

[3.2.2. Chức Năng Hiển Thị Confessions. 17](#_Toc88388326)

[3.2.3. Chức Năng Thêm, Sửa Confessions. 17](#_Toc88388327)

[3.2.4. Chức Năng Xóa Confessions. 18](#_Toc88388328)

[3.2.5. Loading Khi Xử Lý. 19](#_Toc88388329)

[3.2.6. Tìm Kiếm Bài Viết. 19](#_Toc88388330)

[3.2.7. Trang Chi Tiết Bài Viết. 20](#_Toc88388331)

[3.2.7. Đa Ngôn Ngữ. 21](#_Toc88388332)

[3.2.8. Trang Profile Người Dùng. 22](#_Toc88388333)

# 1. PHP Unit Test

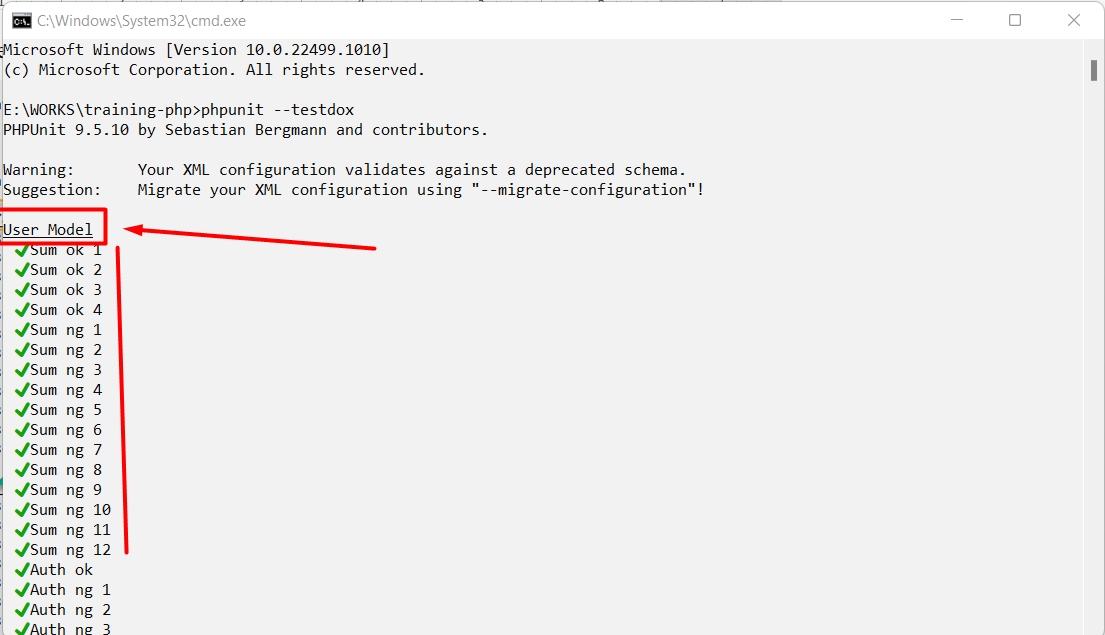
Phân chia công việc:

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **File** | **Function** | **Phân Công** |
| 1 | UserModel.php | sumb | Trần Trọng Hiếu |
| 2 | UserModel.php | auth |
| 3 | UserModel.php | getUsers |
| 4 | UserModel.php | insertUser |
| 5 | UserModel.php | updateUser |
| 6 | UserModel.php | deleteUserById | Ngô Bảy Hên |
| 7 | UserModel.php | findUser |
| 8 | UserModel.php | findUserById |
| 9 | BankModel.php | getBanks |
| 10 | BankModel.php | insertBank | Nguyễn Huỳnh Quốc Trung |
| 11 | BankModel.php | updateBank |
| 12 | BankModel.php | deleteBankById |
| 13 | BankModel.php | findBankById |

## 2.1. Function Sumb.

Bảng mô tả test case:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Function** | **Function Test** | **Kết quả** | **Mô tả** |
| 1 | sumb | testSumOk1 | OK | Cộng số dương và số dương |
| 2 | sumb | testSumOk2 | OK | Cộng số âm và số âm |
| 3 | sumb | testSumOk3 | OK | Cộng số dương và số âm |
| 4 | sumb | testSumOk4 | OK | Cộng hai số thập phân |
| 5 | sumb | testSumNg1 | OK | Cộng số và chuỗi |
| 6 | sumb | testSumNg2 | OK | Cộng chuỗi và chuỗi |
| 7 | sumb | testSumNg3 | OK | Cộng số và null |
| 8 | sumb | testSumNg4 | OK | Cộng null và null |
| 9 | sumb | testSumNg5 | OK | Cộng số và chuỗi rổng |
| 10 | sumb | testSumNg6 | OK | Cộng chuỗi rổng và chuỗi rỗng |
| 11 | sumb | testSumNg7 | OK | Cộng số và mảng |
| 12 | sumb | testSumNg8 | OK | Cộng mảng và mảng |
| 13 | sumb | testSumNg9 | OK | Cộng số và object |
| 14 | sumb | testSumNg10 | OK | Cộng object và object |
| 15 | sumb | testSumNg11 | OK | Cộng số và (true/false) |
| 16 | sumb | testSumNg12 | OK | Cộng (true/false) và (true/false) |

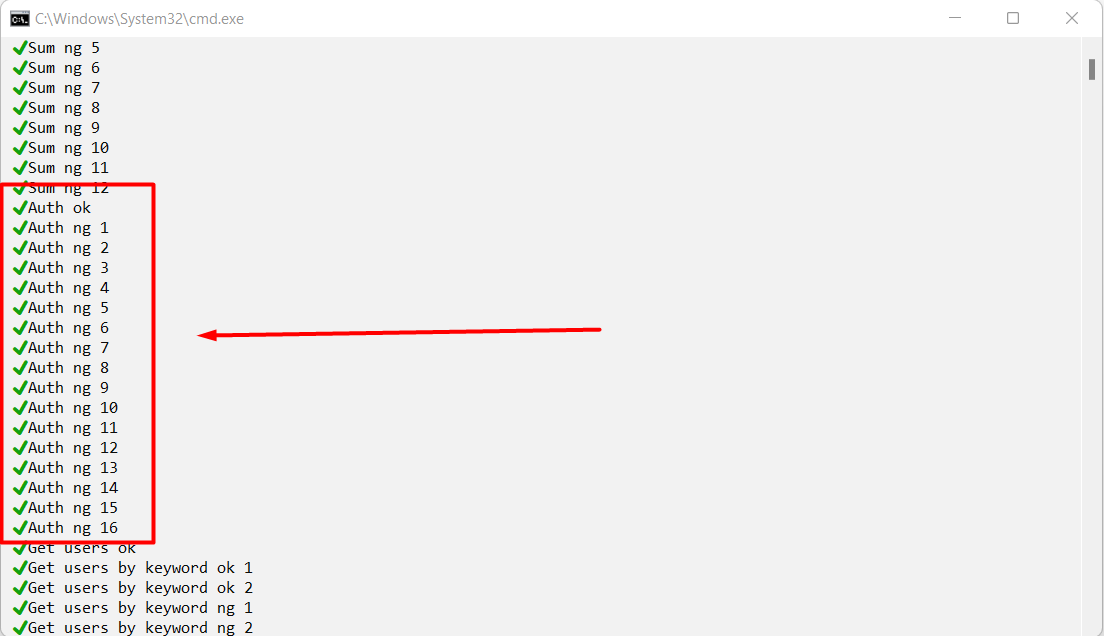


Hình 1. Testdox cho function sumb.

## 2.2. Function Auth.

Bảng mô tả test case:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Function** | **Function Test** | **Kết quả** | **Mô tả** |
| 1 | auth | testAuthOk | OK | Kiểm tra xác thực user |
| 2 | auth | testAuthNg1 | OK | Kiểm tra xác thực user khi username không tồn tại |
| 3 | auth | testAuthNg2 | OK | Kiểm tra xác thực user khi username là rỗng |
| 4 | auth | testAuthNg3 | OK | Kiểm tra xác thực user khi username là null |
| 5 | auth | testAuthNg4 | OK | Kiển tra xác thực user khi username là số |
| 6 | auth | testAuthNg5 | OK | Kiển tra xác thực user khi username là số thực |
| 7 | auth | testAuthNg6 | OK | Kiển tra xác thực user khi username là mảng |
| 8 | auth | testAuthNg7 | OK | Kiển tra xác thực user khi username là object |
| 9 | auth | testAuthNg8 | OK | Kiển tra xác thực user khi username là (true/false) |
| 10 | auth | testAuthNg9 | OK | Kiểm tra xác thực user khi password không chính zác |
| 11 | auth | testAuthNg10 | OK | Kiểm tra xác thực user khi password là rỗng |
| 12 | auth | testAuthNg11 | OK | Kiểm tra xác thực user khi password là null |
| 13 | auth | testAuthNg12 | OK | Kiển tra xác thực user khi password là số |
| 14 | auth | testAuthNg13 | OK | Kiển tra xác thực user khi password là số thực |
| 15 | auth | testAuthNg14 | OK | Kiển tra xác thực user khi password là mảng |
| 16 | auth | testAuthNg15 | OK | Kiển tra xác thực user khi password là object |
| 17 | auth | testAuthNg16 | OK | Kiển tra xác thực user khi password là (true/false) |

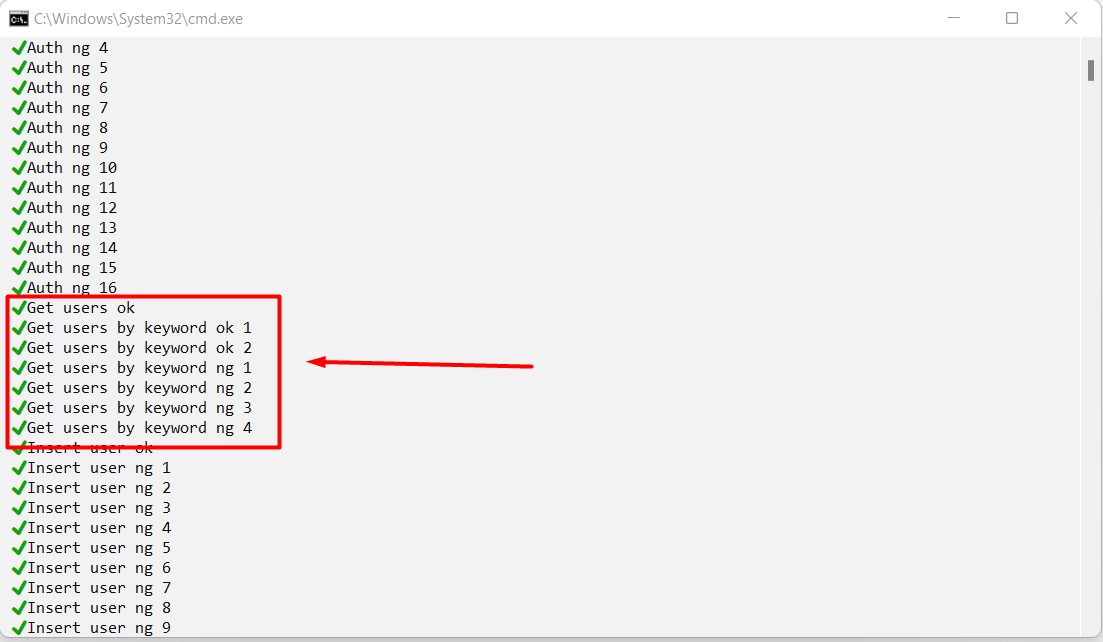


Hình 2. Testdox cho function auth.

## 2.3. Function getUsers.

Bảng mô tả test case:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Function** | **Function Test** | **Kết quả** | **Mô tả** |
| 1 | getUsers | testGetUsersOk | OK | Kiểm tra lấy tất cả users hiện tại |
| 2 | getUsers | testGetUsersByKeywordOk1 | OK | Kiểm tra lấy tất cả users bằng keyword |
| 3 | getUsers | testGetUsersByKeywordOk2 | OK | Kiểm tra lấy tất cả users bằng keyword (chuỗi lạ) |
| 4 | getUsers | testGetUsersByKeywordNg1 | OK | Kiểm tra lấy tất cả users với keyword là số |
| 5 | getUsers | testGetUsersByKeywordNg2 | OK | Kiểm tra lấy tất cả users với keyword là mảng |
| 6 | getUsers | testGetUsersByKeywordNg3 | OK | Kiểm tra lấy tất cả users với keyword là object |
| 7 | getUsers | testGetUsersByKeywordNg4 | OK | Kiểm tra lấy tất cả users với keyword là (true/false) |



Hình 3. Testdox cho function getUsers.

## 2.4. Function insertUser.

Bảng mô tả test case:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Function** | **Function Test** | **Kết quả** | **Mô tả** |
| 1 | insertUser | testInsertUserOk | OK | Kiểm tra thêm user mới |
| 2 | insertUser | testInsertUserNg1 | OK | Kiểm tra thêm user với input là rỗng |
| 3 | insertUser | testInsertUserNg2 | OK | Kiểm tra thêm user với input là null |
| 4 | insertUser | testInsertUserNg3 | OK | Kiểm tra thêm user với input là số |
| 5 | insertUser | testInsertUserNg4 | OK | Kiểm tra thêm user với input là mảng |
| 6 | insertUser | testInsertUserNg5 | OK | Kiểm tra thêm user với input là object |
| 7 | insertUser | testInsertUserNg6 | OK | Kiểm tra thêm user với input là (true/false) |
| 8 | insertUser | testInsertUserNg7 | OK | Kiểm tra thêm user với giá trị key input là rỗng |
| 9 | insertUser | testInsertUserNg8 | OK | Kiểm tra thêm user với giá trị key input là null |
| 10 | insertUser | testInsertUserNg9 | OK | Kiểm tra thêm user với giá trị key input là số |
| 11 | insertUser | testInsertUserNg10 | OK | Kiểm tra thêm user với giá trị key input là mảng |
| 12 | insertUser | testInsertUserNg11 | OK | Kiểm tra thêm user với giá trị key input là object |
| 13 | insertUser | testInsertUserNg12 | OK | Kiểm tra thêm user với giá trị key input là (true/false) |



Hình 4. Testdox cho function insertUser.

## 2.4. Function updateUser.

Bảng mô tả test case:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Function** | **Function Test** | **Kết quả** | **Mô tả** |
| 1 | updateUser | testUpdateUserOk1 | OK | Kiểm tra cập nhật user |
| 2 | updateUser | testUpdateUserOk2 | OK | Kiểm tra cập nhật user với giá trị key input là rỗng |
| 3 | updateUser | testUpdateUserNg1 | OK | Kiểm tra cập nhật user với input là rỗng |
| 4 | updateUser | testUpdateUserNg2 | OK | Kiểm tra cập nhật user với input là null |
| 5 | updateUser | testUpdateUserNg3 | OK | Kiểm tra cập nhật user với input là số |
| 6 | updateUser | testUpdateUserNg4 | OK | Kiểm tra cập nhật user với input là mảng |
| 7 | updateUser | testUpdateUserNg5 | OK | Kiểm tra cập nhật user với input là object |
| 8 | updateUser | testUpdateUserNg6 | OK | Kiểm tra cập nhật user với input là (true/false) |
| 9 | updateUser | testUpdateUserNg7 | OK | Kiểm tra cập nhật user với giá trị key input là null |
| 10 | updateUser | testUpdateUserNg8 | OK | Kiểm tra cập nhật user với giá trị key input là số |
| 11 | updateUser | testUpdateUserNg9 | OK | Kiểm tra cập nhật user với giá trị key input là mảng |
| 12 | updateUser | testUpdateUserNg10 | OK | Kiểm tra cập nhật user với giá trị key input là object |
| 13 | updateUser | testUpdateUserNg11 | OK | Kiểm tra cập nhật user với giá trị key input là (true/false) |
| 14 | updateUser | testUpdateUserNg12 | OK | Kiểm tra cập nhật user với giá trị id input là rỗng |
| 15 | updateUser | testUpdateUserNg13 | OK | Kiểm tra cập nhật user với giá trị id input là null |
| 16 | updateUser | testUpdateUserNg14 | OK | Kiểm tra cập nhật user với giá trị id input là số |
| 17 | updateUser | testUpdateUserNg15 | OK | Kiểm tra cập nhật user với giá trị id input là mảng |
| 18 | updateUser | testUpdateUserNg16 | OK | Kiểm tra cập nhật user với giá trị id input là object |
| 19 | updateUser | testUpdateUserNg17 | OK | Kiểm tra cập nhật user với giá trị id input là (true/false) |



Hình 5. Testdox cho function updateUser.

# 2. DỰ ÁN CHUYÊN ĐỀ WEB 2

Ứng dụng Confessions Vietnam.

Trong thời buổi công nghệ hiện nay, việc chia sẻ thông tin là rất quan trọng. Những vấn đề trong cuộc sống khó khăn càng khiến người ta mệt mỏi và cần người tâm sự. Ứng dụng được tạo ra giúp các bạn chia sẻ và giải tỏa những điều trong lòng, những niềm vui, nỗi buồn một cách hoàn toàn ẩn danh, hoàn toàn bảo mật. Chia sẻ và kết nối cùng những người khác.

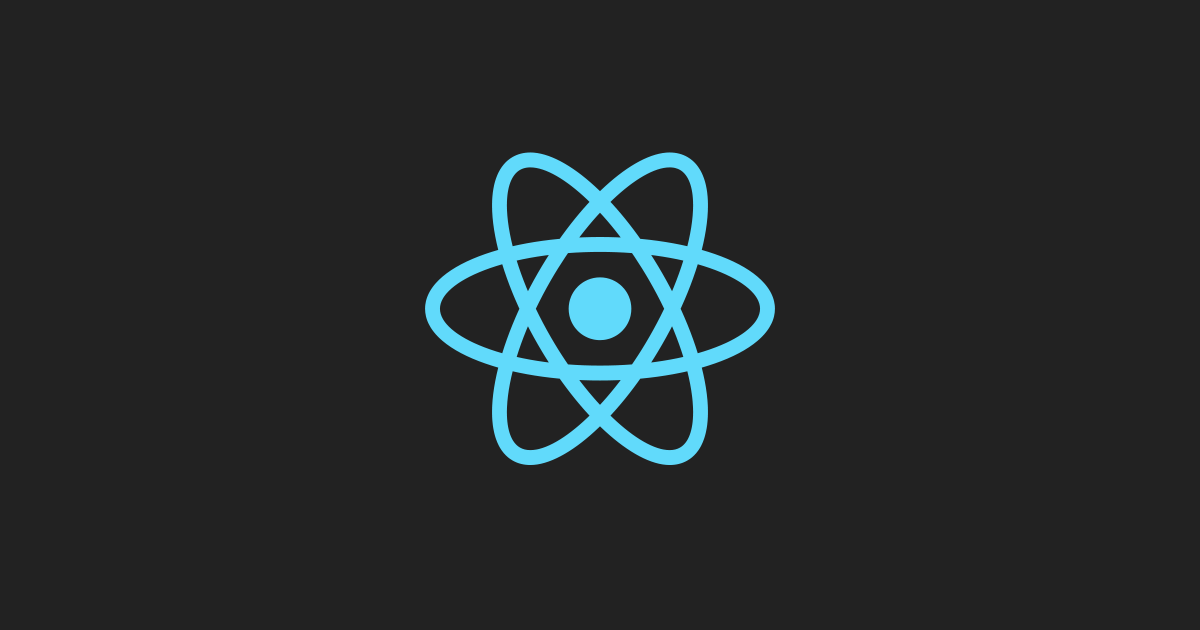
## 3.1. Mô Tả Công Nghệ Được Sử Dụng

Công nghệ được sử dụng: ReactJS (Redux, NextJS)

### 3.1.1. ReactJS là gì?

ReactJs là một thư viện viết bằng **Javascript**, dùng để xây dựng giao diện người dùng (UI). ReactJS được duy trì bởi Facebook và cộng đồng developer hùng mạnh. React cơ bản được sử dụng để phát triển các các **Single Page Application**.

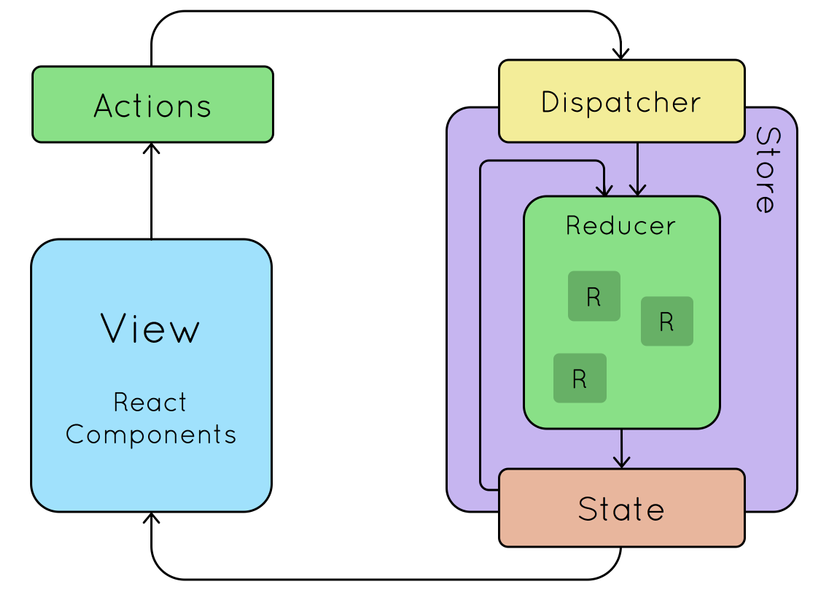
ReactJS sử dụng để tạo giao diện và quản lý cấu trúc cực tốt, và được đứng sau bởi một ông lớn trong ngành IT. Ngoài ra ReactJS được kế thừa một kho thư viện đồ sộ trên NPM. Không thể không kể đến Redux.



*Hình 16. Hình ảnh Logo của ReactJS*

### 3.1.2. Redux là gì?

Redux là một thư viện Javascript của **ReactJS**, giúp tạo ra thành một lớp quản lý state của ứng dụng. Mặc định ReactJS cũng có thể quản lý state chung (**React Context**) nhưng cũng khó có thể thay cho Redux vì khả năng quản lý state rõ ràng của nó.



*Hình 17. Hình ảnh mô tả vòng đời (life cycle) của Redux*

### 3.1.3. NextJS là gì?

NextJS được tạo ra bởi Vercel là một framework của ReactJS được xây dựng trên nền NodeJS (**Server Side Rendering SSR**) khác với ReactJS là **Client Side Rendering CSR** giúp tạo ra một trang viết bằng ReactJS nhưng được render phía Server, giúp tăng cải thiện SEO đối với một ứng dụng ReactJS.



*Hình 18. Hình ảnh Logo của NextJS*

## 3.2. Đặc Tả Yêu Cầu Dự Án

Các chức năng và phân công công việc CDPTW1:

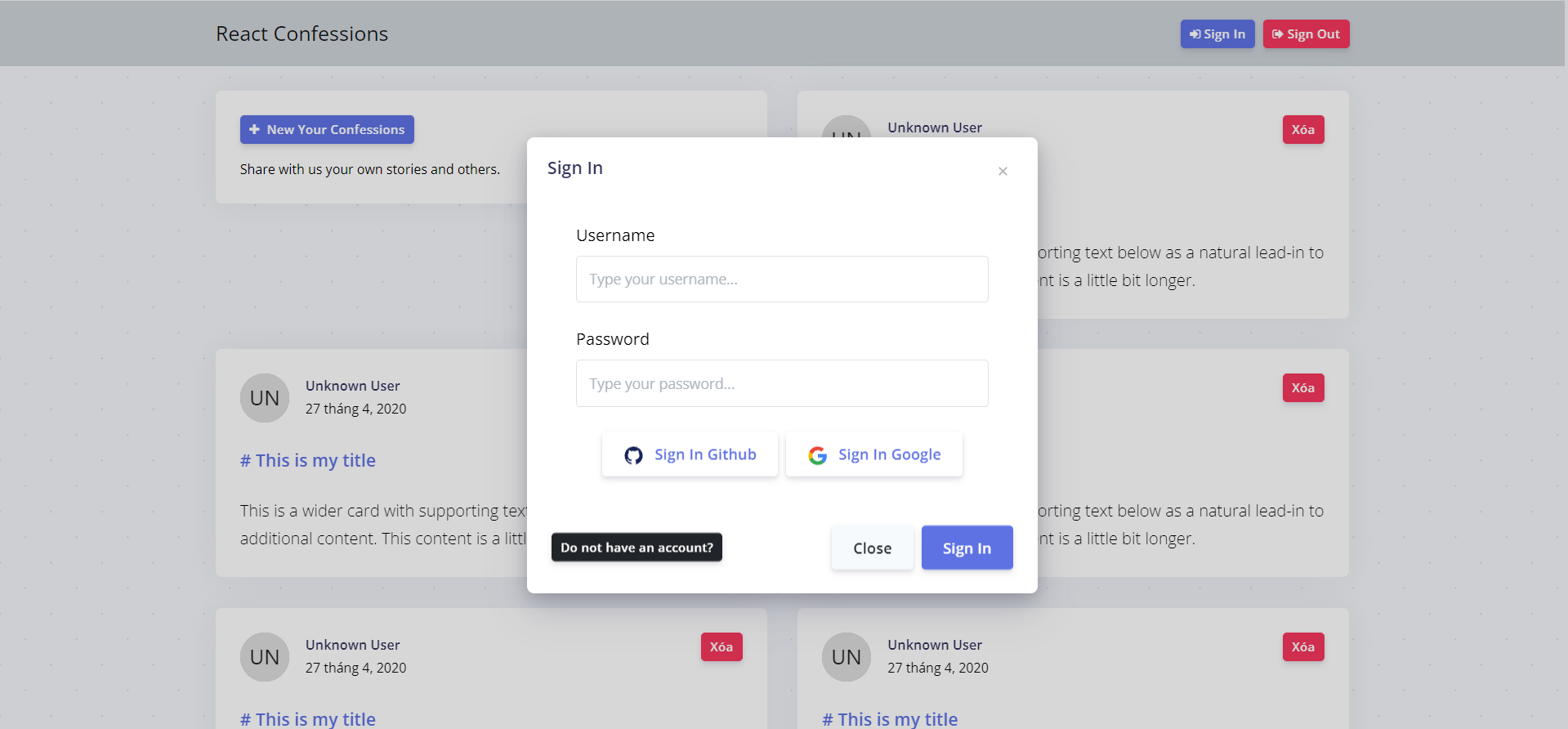
|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Thành Viên** | **Nội Dung Được Phân Công** | **Hoàn Thành** |
| 1 | Ngô Bảy Hên | Đăng nhập,  Đăng ký,  Đăng xuất (1) | 100% |
| 2 | Trần Trọng Hiếu | Sửa confessions (3),  Hiển thị confessions,  Phân trang confessions (2) | 100% |
| 3 | Nguyễn Huỳnh Quốc Trung | Thêm confessions (3),  Xóa confessions (4) | 100% |

Các chức năng và phân công công việc CDPTW2:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Thành Viên** | **Nội Dung Được Phân Công** | **Thời Gian** | **Hoàn Thành** |
| 1 | Ngô Bảy Hên | Loading khi xử lý API (5),  Hiển thị trang chi tiết bài viết (bài viết liên quan) (7) | 10 ngày | 10% |
| 2 | Trần Trọng Hiếu | Đa Ngôn Ngữ (8),  Hiển thị trang profile người dùng (9) | 14 ngày | 10% |
| 3 | Nguyễn Huỳnh Quốc Trung | Tìm kiếm bài viết (6),  Hiển thị trang chi tiết bài viết (nội dung) (7) | 10 ngày | 10% |

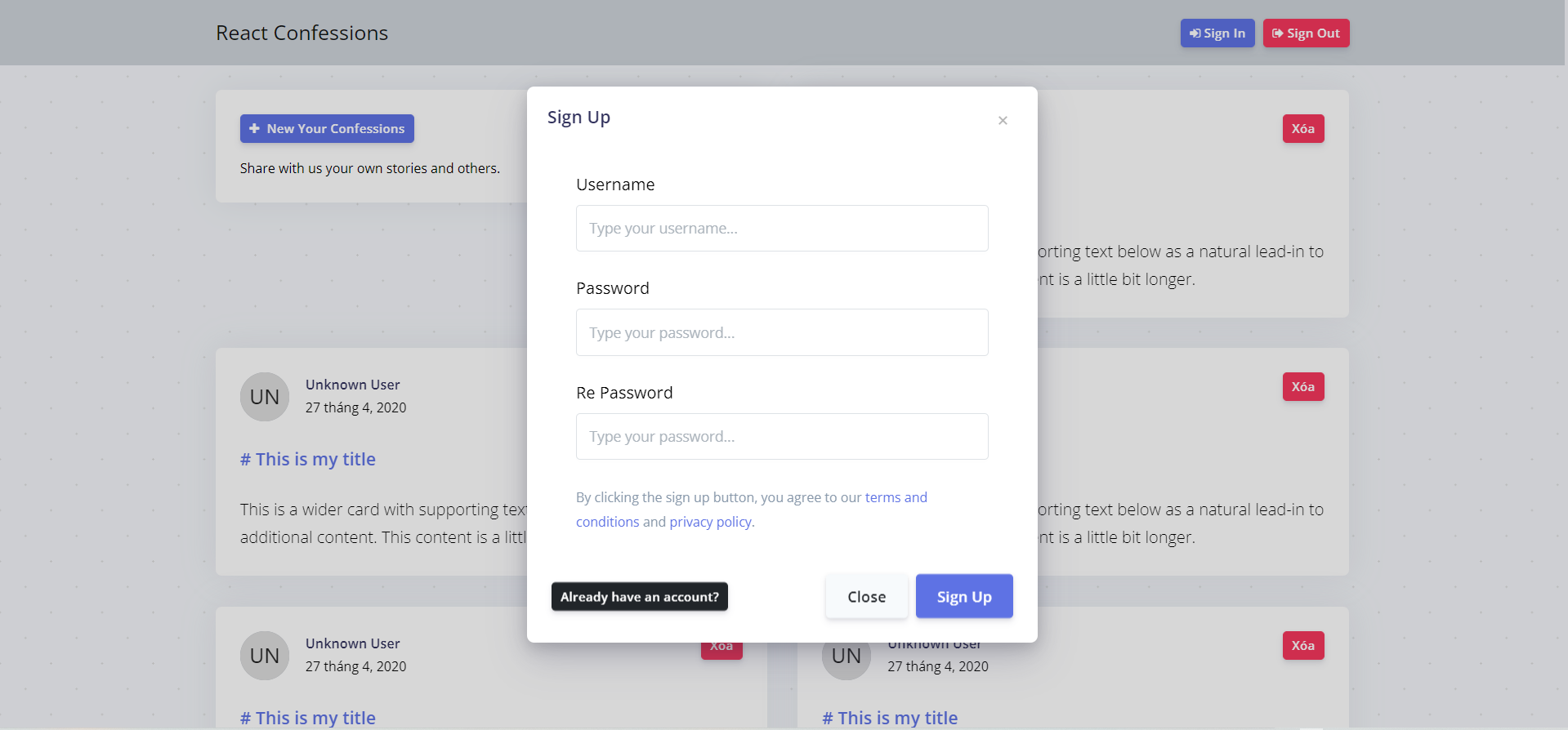
### 3.2.1. Chức Năng Đăng Nhập, Đăng Ký, Đăng Xuất

Hiển thị nút đăng nhập và đăng xuất ở menu, và mở modal form đăng nhập khi người dùng nhấp vào nút đăng nhập.



*Hình 19. Giao Diện Chức Năng Đăng Nhập*

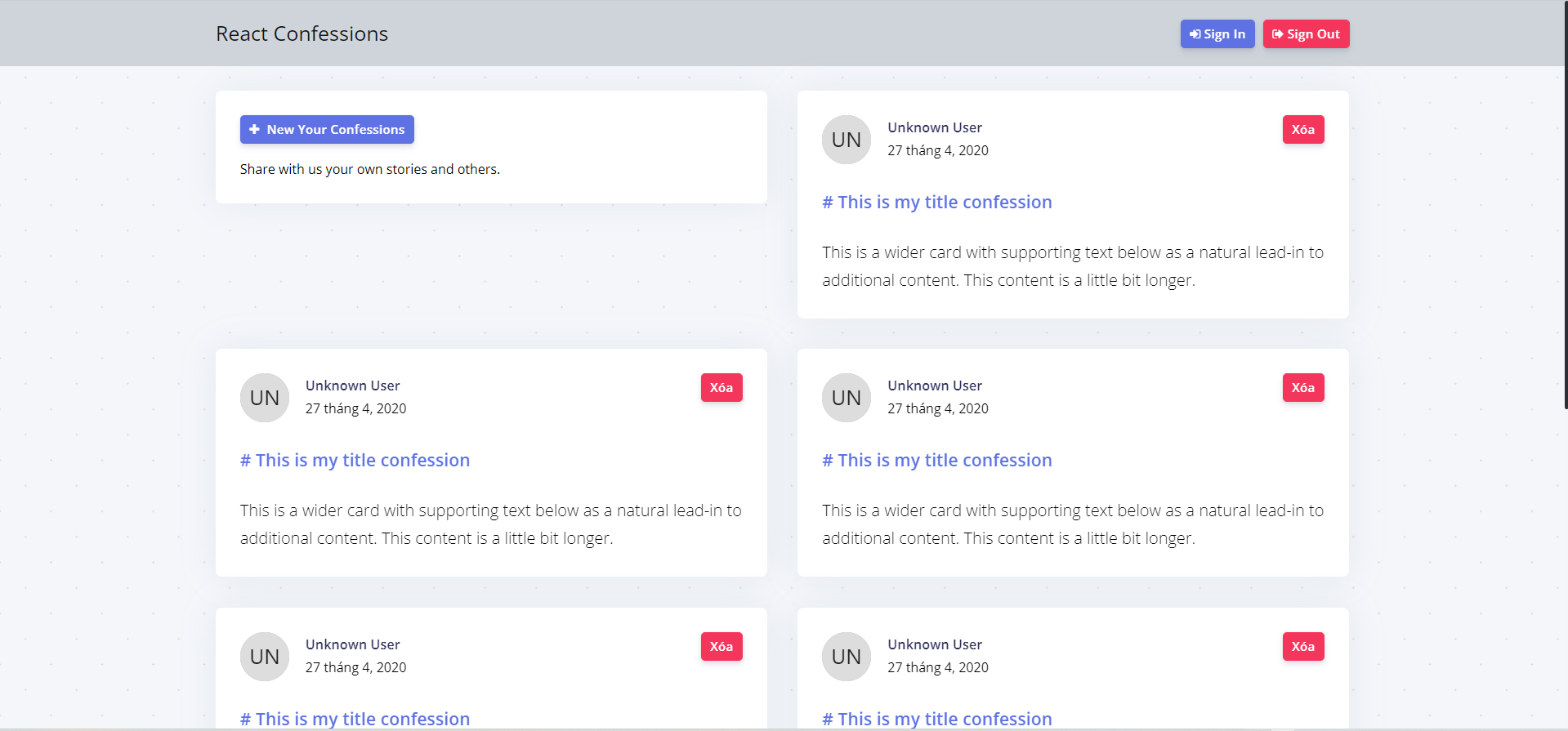
Nếu người dùng chưa có tài khoản, khi nhấn vào nút “Do not have an account?”, giao diện sẽ chuyển sang modal đăng ký.



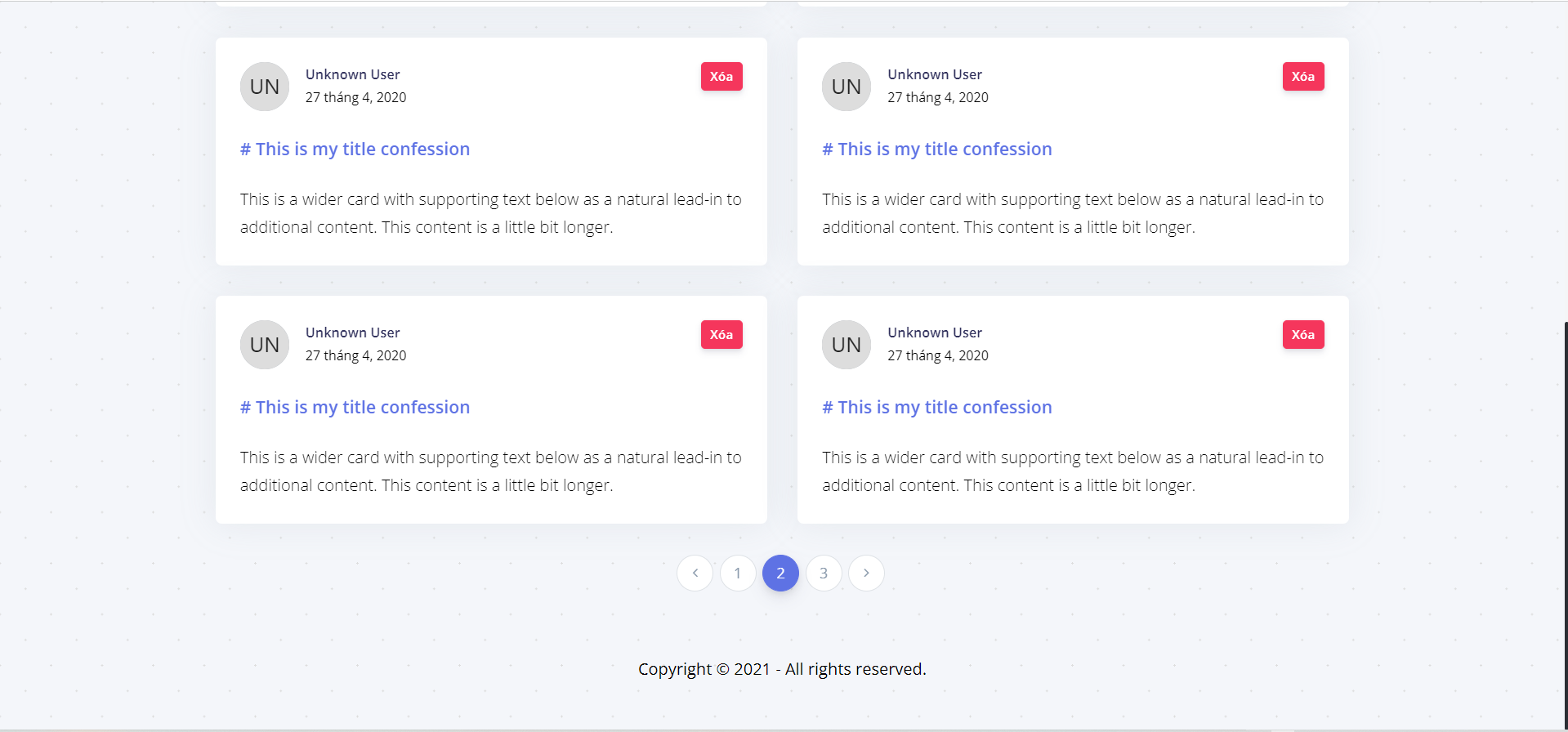
*Hình 20. Giao Diện Chức Năng Đăng Ký*

### 3.2.2. Chức Năng Hiển Thị Confessions.

Hiển thị toàn bộ confessions của người dùng đăng lên (dạng ẩn danh, không chứa tên) danh sách hiển thị mới nhất đến cũ nhất. Hiển thị phân trang.



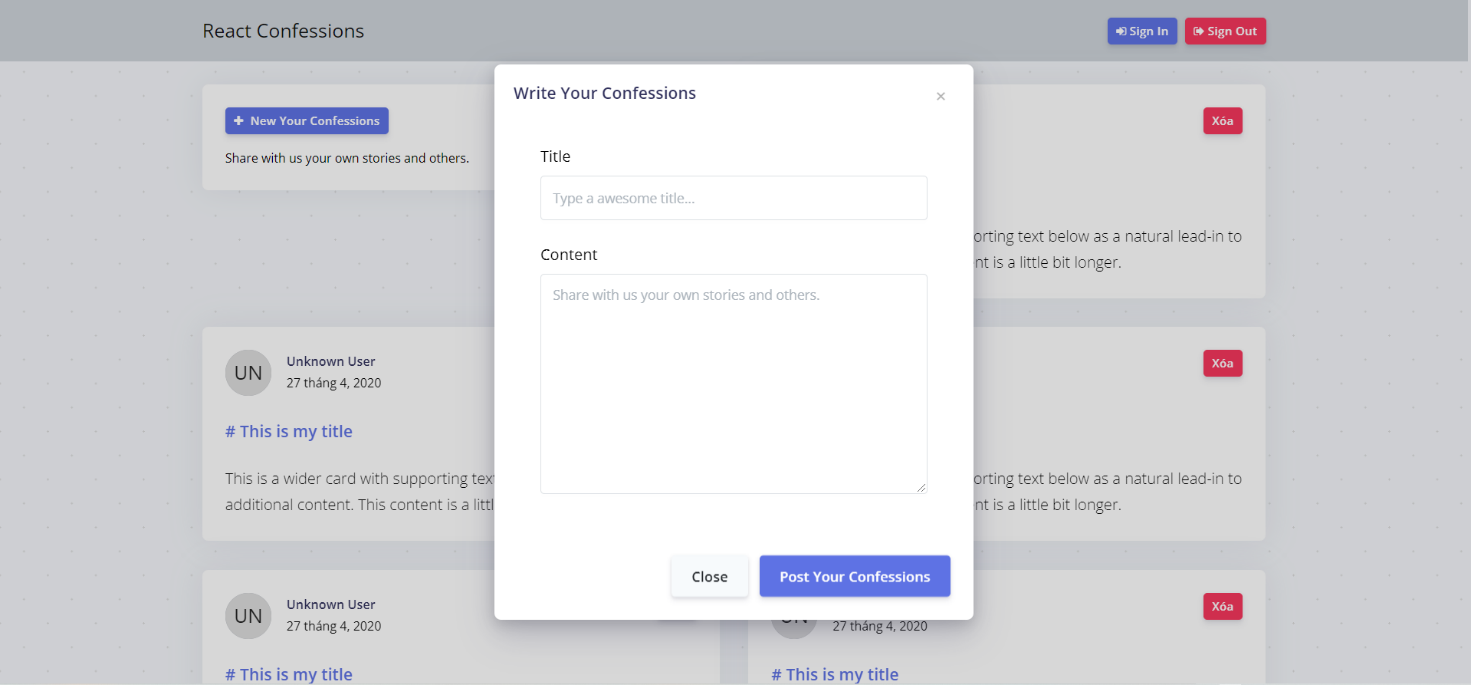
*Hình 21. Giao Diện Hiển Thị Confessions*



*Hình 22. Giao Diện Phân Trang Hiển Thị Confessions*

### 3.2.3. Chức Năng Thêm, Sửa Confessions.

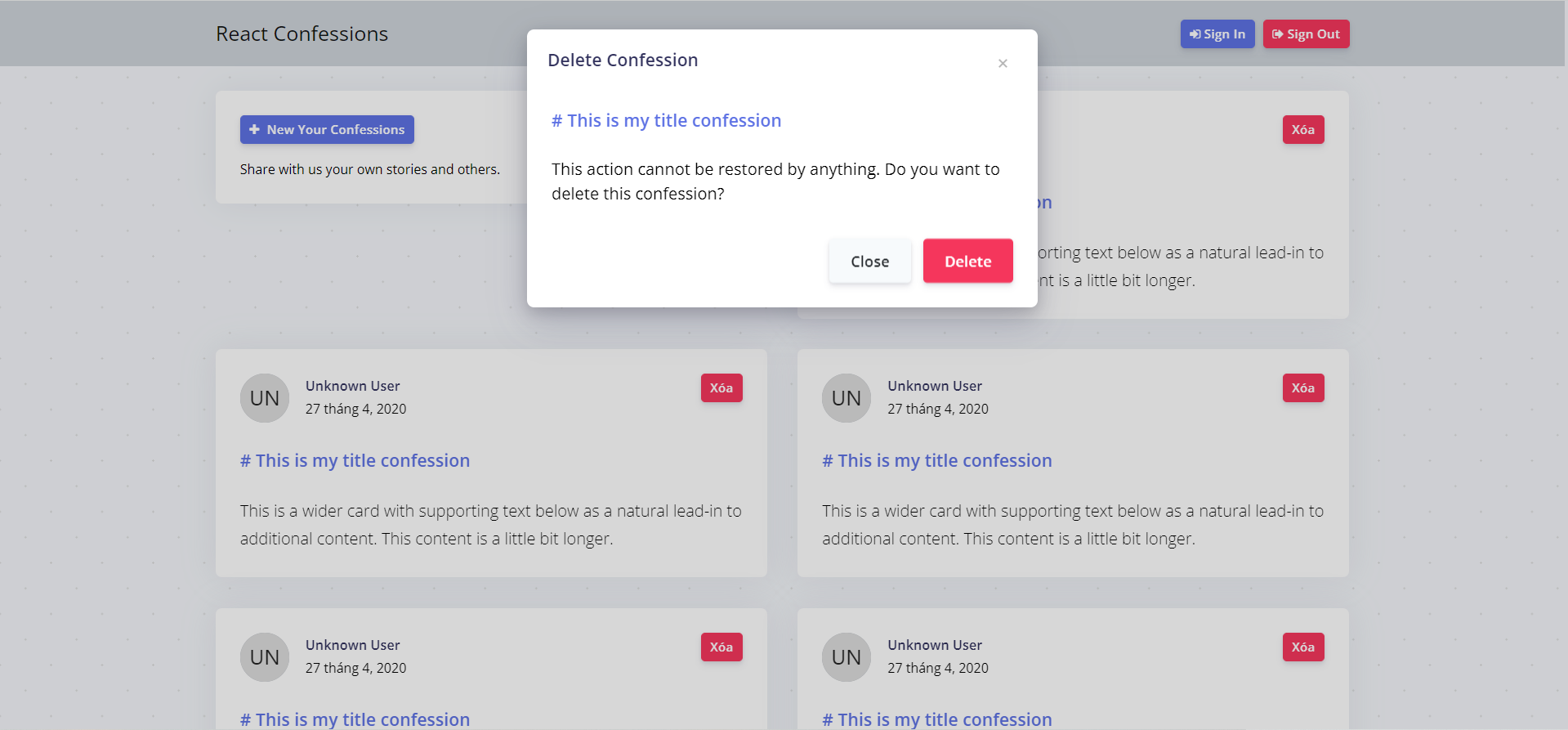
Hiển thị modal form thêm và sửa confession khi người dùng nhấn vào nút “New Your Confessions”, bao gồm tiêu đề và nội dung của confession. Người dùng chỉ sửa được bài viết khi tài khoản đăng nhập.



*Hình 23. Giao Diện Thêm Và Sửa Confessions*

### 3.2.4. Chức Năng Xóa Confessions.

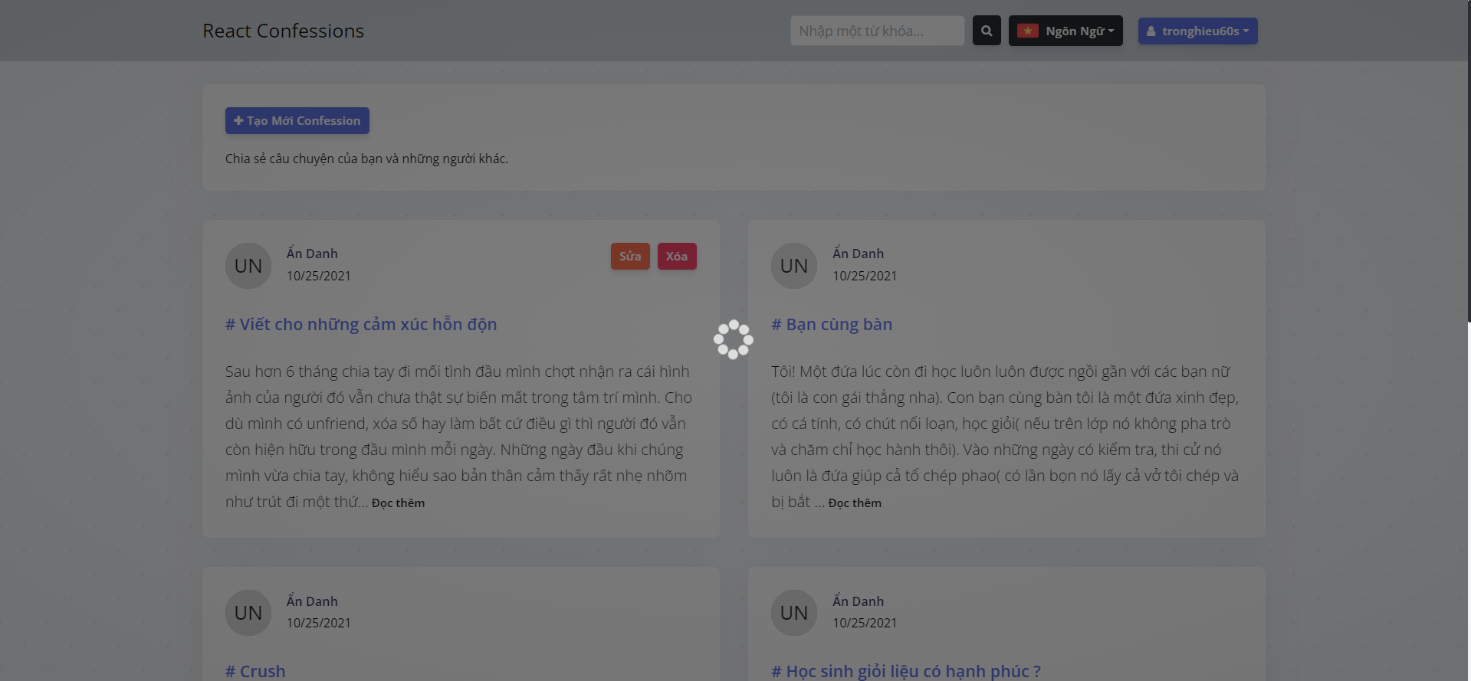
Hiển thị modal xác nhận muốn xóa confessions khi người dùng bấm vào nút xóa. Chỉ hiển thị khi người dùng đã đăng nhập.



*Hình 24. Giao Diện Xóa Confessions*

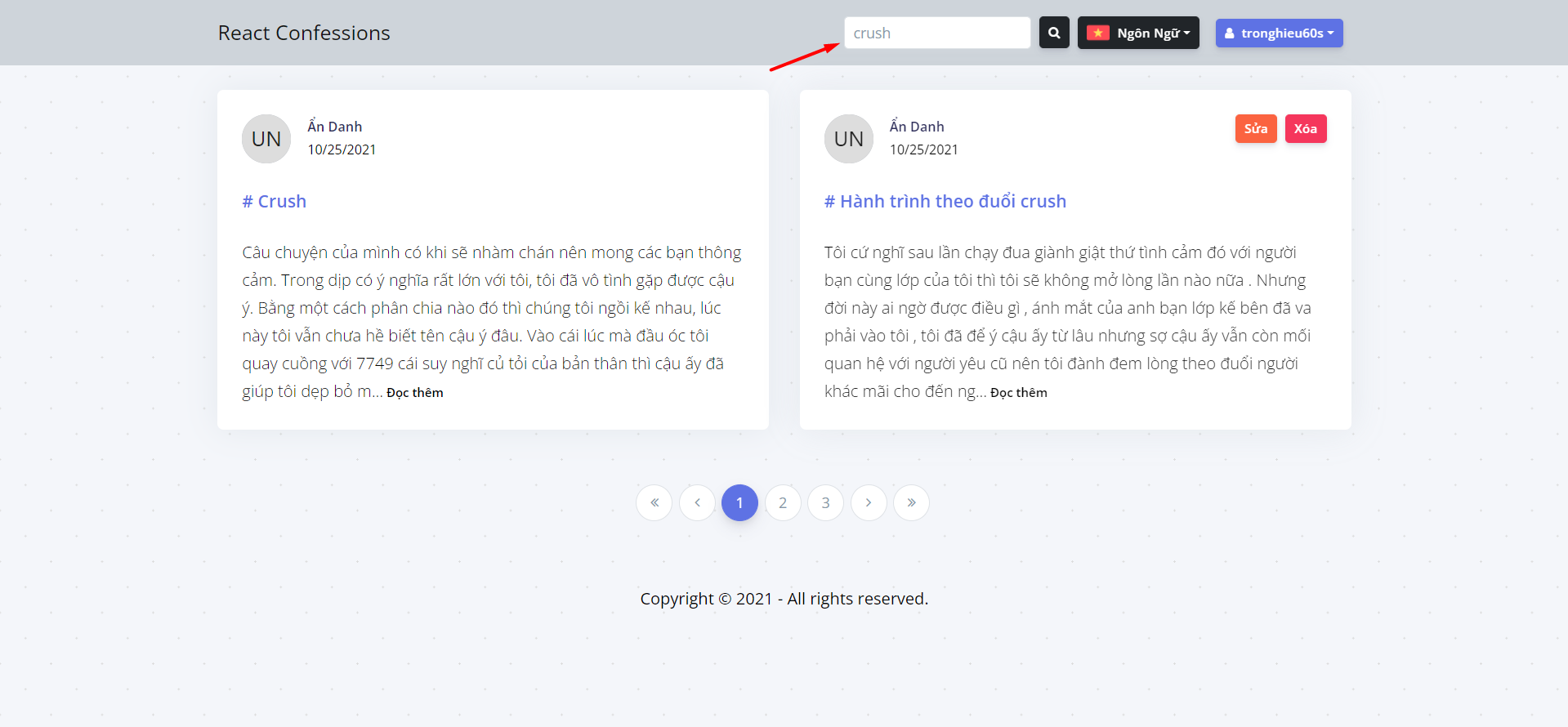
### 3.2.5. Loading Khi Xử Lý.

Loading sẽ được hiện lên khi người dùng có hành động xử lý và gọi đến API của ứng dụng, tăng UI/UX người dùng.



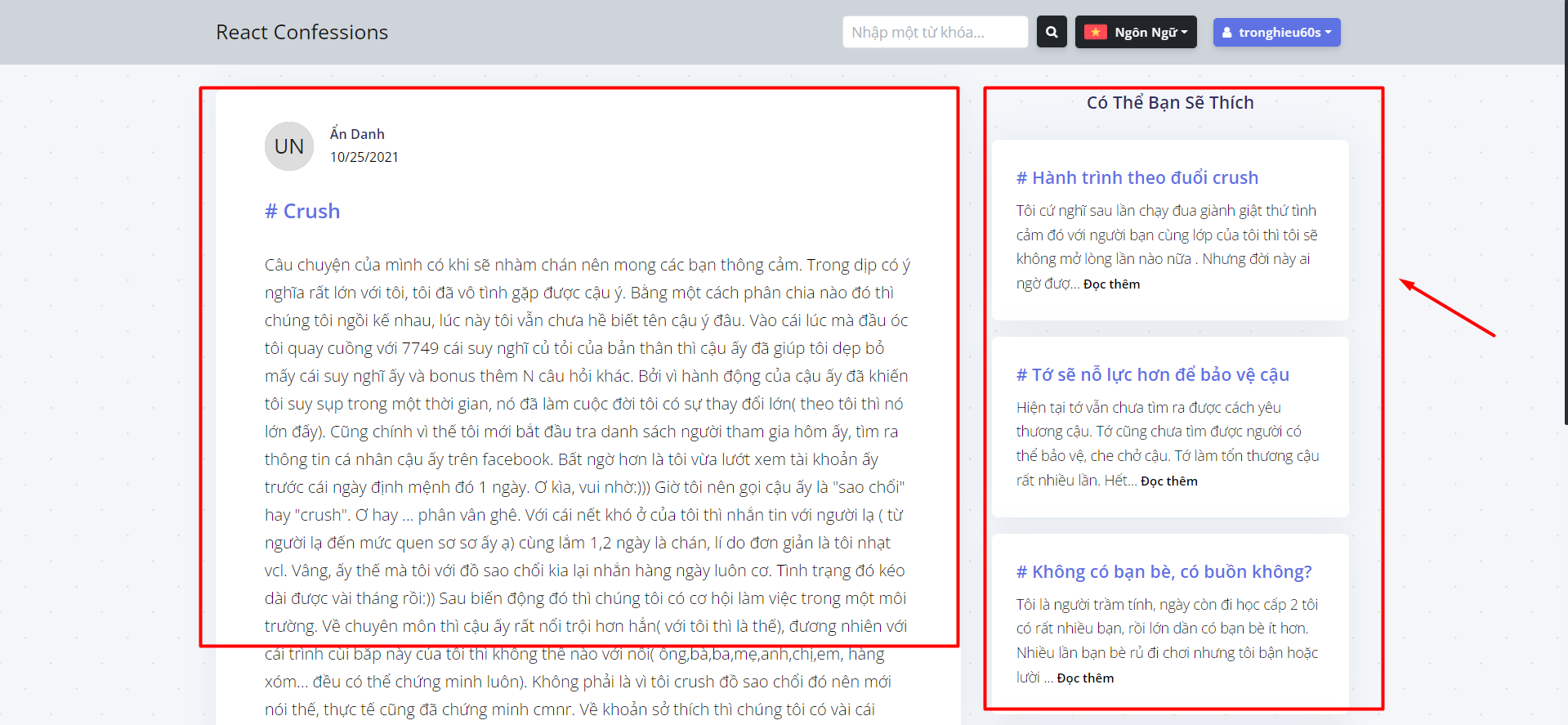
### 3.2.6. Tìm Kiếm Bài Viết.

Người dùng có thể tìm kiếm được cái bài viết trong ứng dụng nhanh chóng.



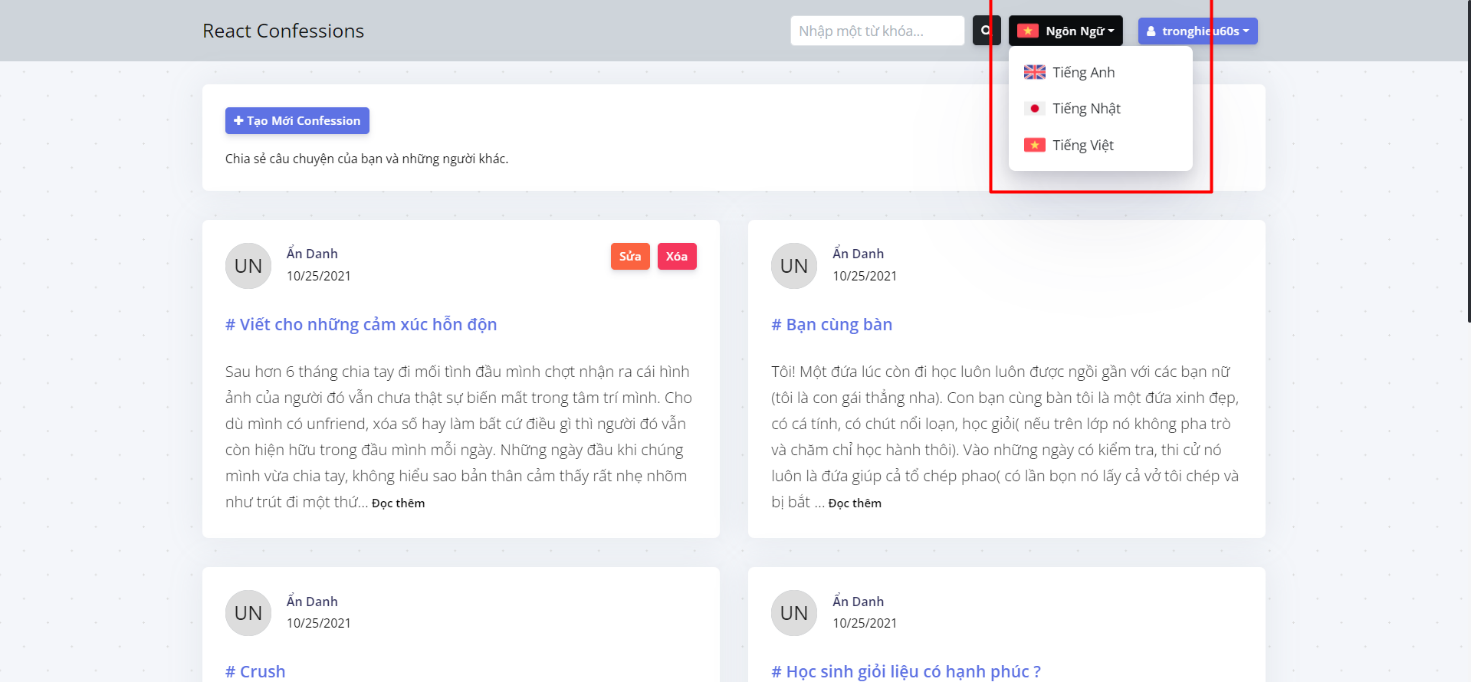
### 3.2.7. Trang Chi Tiết Bài Viết.

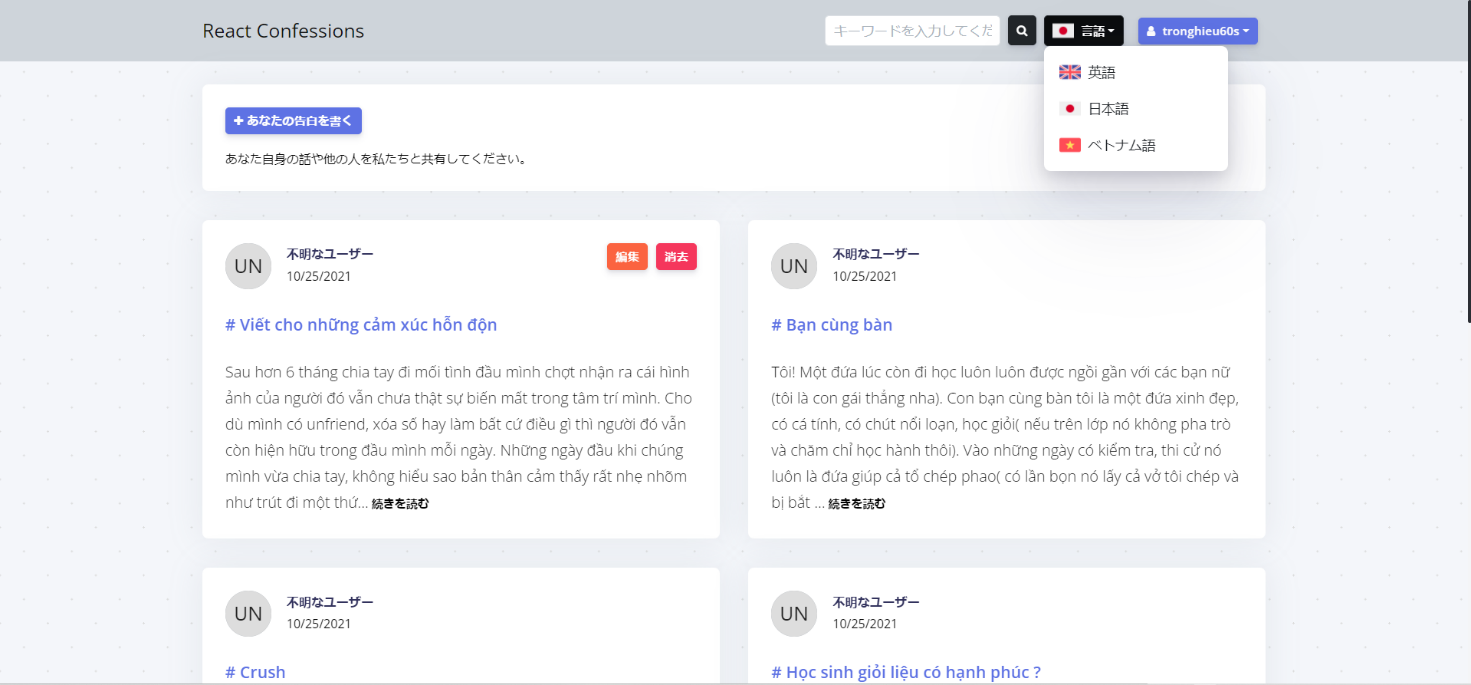
Trang chi tiết hiển thị nội dung bài viết ở bên trái, và nội dung bài viết liên quan ở bên phải, người dùng có thể click vào và đọc tiếp các bài viết thú vị khác.



### 3.2.7. Đa Ngôn Ngữ.

Người dùng có thể đổi một số ngôn ngữ khác, để phù hợp với các vùng khác nhau, hiện tại đang hỗ trợ Anh, Việt, Nhật.





### 3.2.8. Trang Profile Người Dùng.

Hiển thị chi tiết người dùng, các bài viết mà người dùng đã tạo, có thể chỉnh sửa ngay. Hình ảnh người dùng được lấy từ email của gavatar.

