**TRƯỜNG ĐẠI HỌC PHENIKAA**

**KHOA CÔNG NGHỆ THÔNG TIN**

****

**Báo cáo dự án môn lập trình web nâng cao**

**Đề tài: “Website quản lý tài sản”**

|  |  |
| --- | --- |
| **Giảng viên hướng dẫn: Nguyễn Thành Trung**  **Sinh viên thực hiên: Nguyễn Văn Tuấn** |  |

*Hà Nội, ngày 22 tháng 03 năm 2022*

**Mục lục**

[**I, Giới thiệu chung** 3](#_Toc99377363)

[**II, Mô tả về hệ thống** 3](#_Toc99377364)

[**1, Mục tiêu** 3](#_Toc99377365)

[**2, Các công nghệ, thư viện sử dụng** 4](#_Toc99377366)

[**3, Các usecase và features chính** 5](#_Toc99377367)

[**4, Thiết kế** 7](#_Toc99377368)

[**1, Data modeling** 7](#_Toc99377369)

[**2, Kiến trúc hệ thống** 7](#_Toc99377370)

[**3, Giao diện hệ thống** 8](#_Toc99377371)

# **I, Giới thiệu chung**

* **Thực trang**: Nhận thấy mỗi khi vào làm một công ty nào đó, thường thì người đó sẽ được công ty cấp cho màn hình, Pc, bàn phím và một số trang thiết bị cần thiết cho công việc. Khi mà số lượng nhân viên đông lên, người xin vào làm, người xin ra nhiều hơn sẽ dẫn đến việc kiểm soát các vật tư, tài sản của công tư sẽ phức tạp và khó khăn hơn.
* **Biện pháp**: Chính vì vậy em đã lên ý tưởng một website giúp cho việc quản lý tài sản công ty trở lên đơn giản hơn.
* **Source Code**: <https://github.com/TuanPointC/AssetManager.git>

# **II, Mô tả về hệ thống**

## **1, Mục tiêu**

* Tạo một website thân thiện dễ dàng sử dụng.
* Thực hiện cơ bản, đầy đủ các chức năng để quản lý tài sản dễ dàng:

+ Quản lý người dùng, sự phân quyền rõ ràng giữa quản trị viên và người nhân viên.

+ Quản lý tài sản của công ty.

+ Quản lý việc bàn giao, thu hồi tài sản đối với nhân viên.

+ Chức năng trả tài sản đối với nhân viên

+ Thông kê tài sản.

* Code sạch, dễ bảo trì, nâng cấp sau này.
* Tự vận dụng các kiến thức tự học để xây dựng hệ thống.

## **2, Các công nghệ, thư viện sử dụng**

- Website xây dựng dựa trên mô hình Singe Page Application hay Single Page Web Application là một ứng dụng web hay một website mà ở đó tất cả các thao tác của người dùng chỉ diễn ra trên 1 trang duy nhất, tất cả các cấu trúc của trang web (HTML) được load sẵn 1 lần và sẽ không load lại ngay cả khi chuyển trang.

Một vài website nổi tiếng như: Gmail, Facebook, Youtube, Twitter, ... đều đang sử dụng SPA để tạo nên những trải nghiệm mang tính chiều sâu và chiều rộng cho người dùng. Có rất nhiều lợi ích khi sử dụng giải pháp SPA như cải thiện hiệu năng và tính nhất quán, giảm thiểu thời gian phát triển và chi phí hạ tầng.

Khi phân chia sự trình bày nội dung (frontend) và dữ liệu (backend), đội phát triển sẽ có thể làm việc với tốc độ song song nhưng vẫn liên kết đến giải pháp chung. SPA vẫn tốt cho việc responsive từ mobile, tablet cho đến desktop.

- Backend sử dụng C# theo .Net core framework, đây là một framework được Microsoft phát triển. Là một nền tảng phát triển đa mục đích. .Net Core là nền tảng chéo (hỗ trợ Windows, macOS và Linux) và có thể được sử dụng để xây dựng các ứng dụng thiết bị, đám mây và IoT.

- FrontEnd sử dụng React là thư viện Javascript phổ biến nhất để xây dựng giao diện người dùng (UI). Nó cho tốc độ phản hồi tuyệt vời khi user nhập liệu bằng cách sử dụng phương pháp mới để render trang web. Components của công cụ này được phát triển bởi Facebook. Nó được ra mắt như một công cụ JavaScript mã nguồn mở vào năm 2013. Hiện tại, nó đã đi trước các đối thủ chính như Angular và Vuejs, hai thư viện JavaScript bán chạy nhất thời bấy giờ.

- Database sử dụng SQL server một phần mềm do Microsoft phát triển vào năm 1989, là hệ thống quản trị cơ sở dữ liệu quan hệ hay còn gọi là RDBMS (Relational Database Management System). SQL Server được dùng vào việc tạo, duy trì và lưu trữ, phân tích và trích xuất cơ sở dữ liệu theo yêu cầu.

- Một số thư viện nổi bật được dùng:

+ Swagger: là một bộ công cụ mã nguồn mở để xây dựng OpenAPI specifications giúp chúng ta có thể thiết kế, xây dựng tài liệu và sử dụng REST APIs.

+ Entity framework: giúp các nhà phát triển Web tương tác với dữ liệu quan hệ theo phương pháp hướng đối tượng với ít mã hơn so với các ứng dụng truyền thống. Lợi ích lớn nhất của nó là giúp lập trình viên giảm thiểu việc lập trình mã nguồn để thực hiện truy cập và tương tác với cơ sở dữ liệu.

+ Axios : Axios là một HTTP client được viết dựa trên Promises được dùng để hỗ trợ cho việc xây dựng các ứng dụng API từ đơn giản đến phức tạp và có thể được sử dụng cả ở trình duyệt hay Node. Js.

+ Antdesign: Ant là tập hợp các components của React được xây dựng theo chuẩn thiết kế của Ant UED Team. Tương tự như chuẩn Material Design, Ant cung cấp hầu hết các component thông dụng trong ứng dụng web hiện đại, như Layout, Button, Icon, DatePicket.

+ Ngoài ra dung một sốứng dụng, thư viện khác giúp việc code dễ dàng, nhanh chóng hơn: Postman, Visual code, visual studio code,…

## **3, Các usecase và features chính**

* Xác thực

+ Chức năng đăng nhập

+ Chức năng đăng xuất

+ Chức năng đổi mật khẩu

* Quản lý User

+ Hiển thị danh sách user, tìm kiếm, lọc user

+ Thêm người dùng

+ Chỉnh sửa người dùng

+ Xóa người dùng

* Quản lý Tài sản

+ Hiển thị danh sách tài sản, tìm kiếm, lọc tài sản

+ Thêm tài sản

+ Chỉnh sửa tài sản

+ Xóa tài sản

* Quản lý việc bàn giao tài sản

+ Tạo bàn giao tài sản cho user

+ Hiện thị các assignments đang có

+ Chỉnh sửa các assignment

+ Xóa các assignment

+ Chấp nhận hay từ chối bàn giao của nhân viên

* Quản lý việc yêu cầu trao trả tài sản

+ Tạo yêu cầu hoàn trả tài sản

+ Duyệt yêu cầu bàn giao tài sản

* Quản lý thống kê tài sản

+ Thông kê theo đặc điểm tài sản

+ Xuất bảng thống kê ra excel.

## **4, Thiết kế**

### **1, Data modeling**

Diagram

Description automatically generated

### 

### **2, Kiến trúc hệ thống**

* Kiến trúc

Diagram

Description automatically generated

* Biểu đồ luồng xử lý

+ Login

Diagram

Description automatically generated

+ Logout

Diagram

Description automatically generated

+ Change Password

Diagram

Description automatically generated

Diagram

Description automatically generated+ Quản lý User

+ Quản lý tài sản

Diagram

Description automatically generated

Diagram

Description automatically generated+ Quản lý Assignment

Diagram

Description automatically generated+ Quản lý thống kê

Diagram

Description automatically generated+ Quản lý Request for returning

### **3, Giao diện hệ thống**

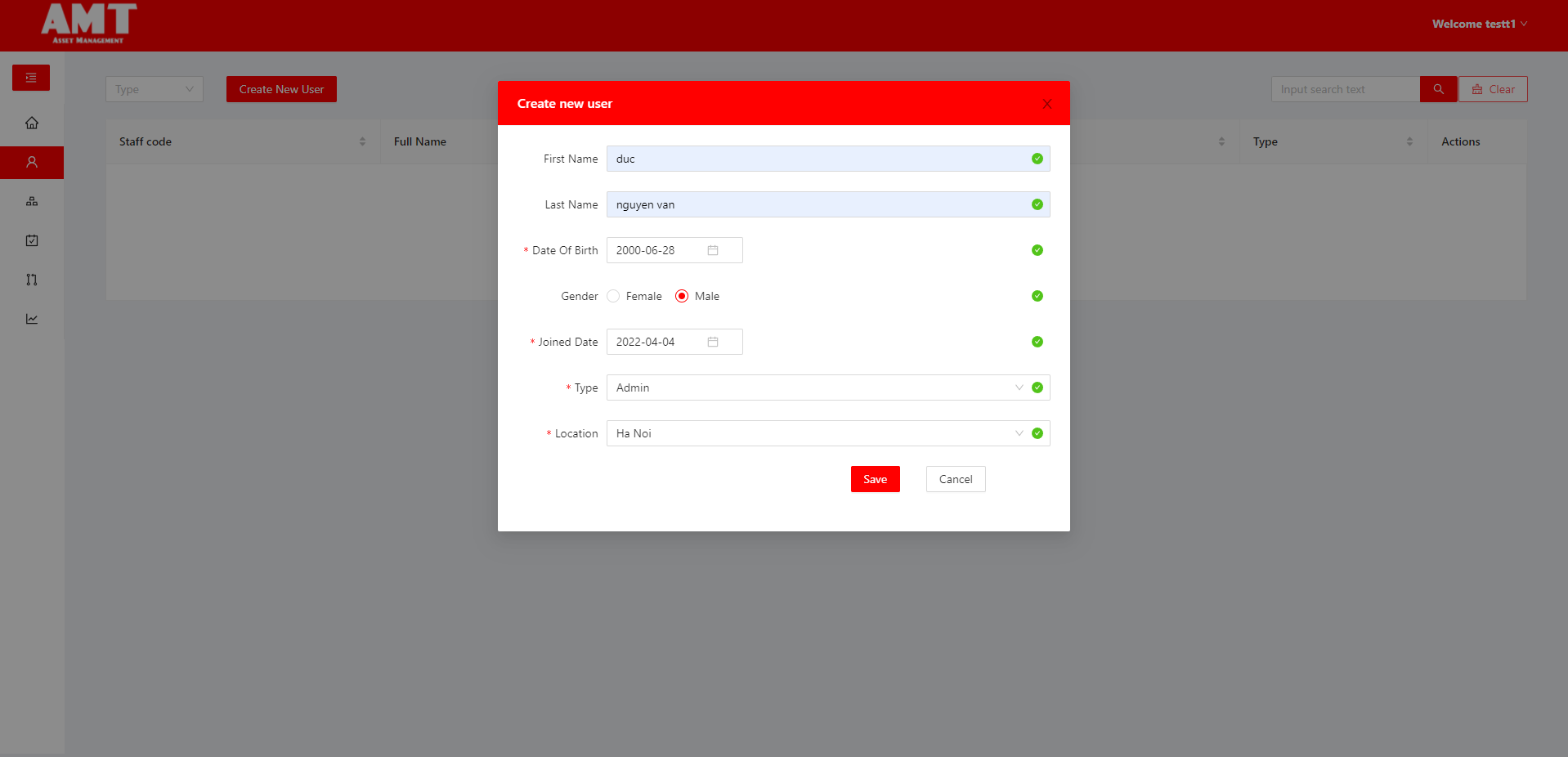


Figure 1 tạo tài khoản mới

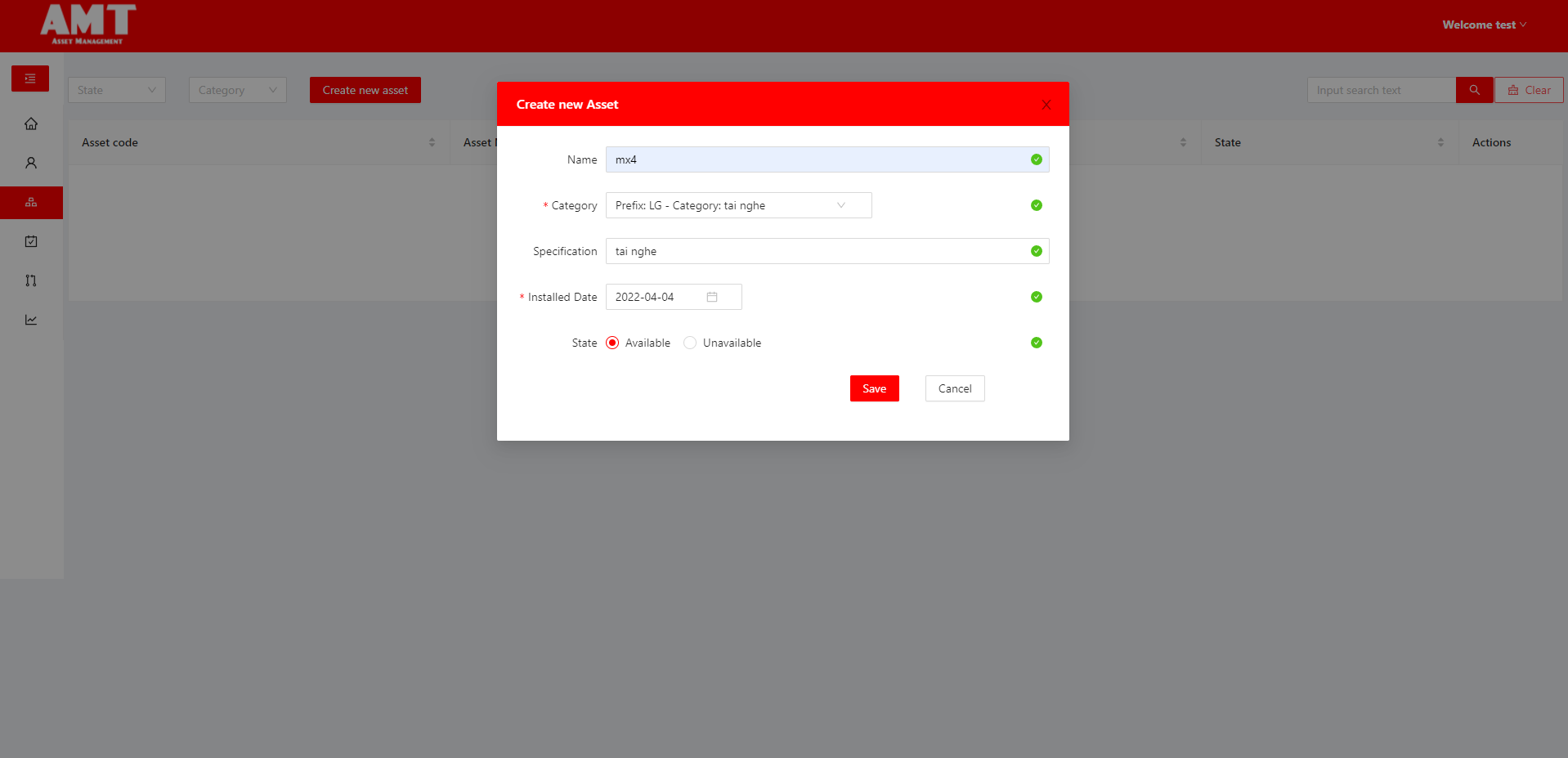
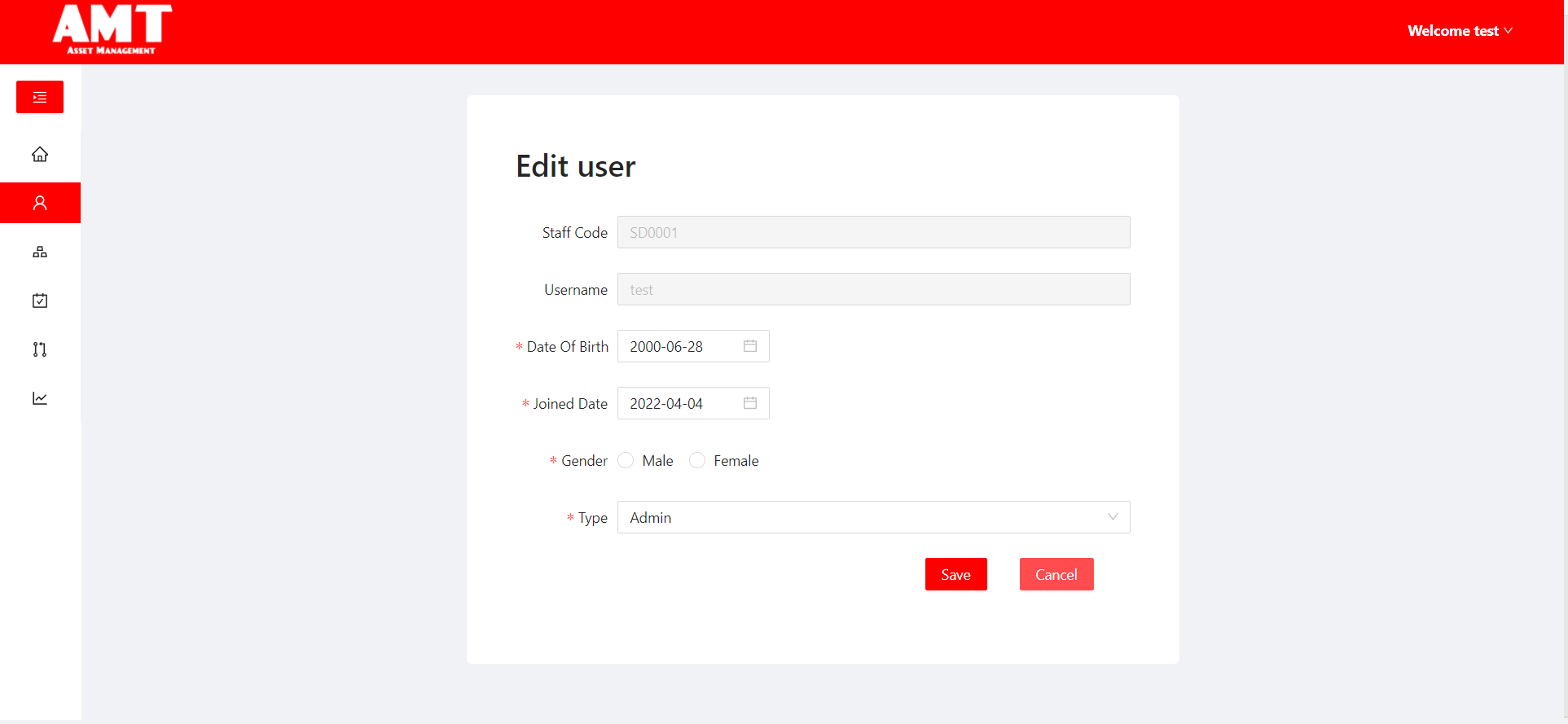
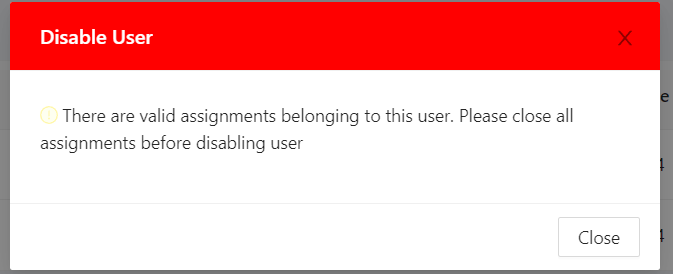
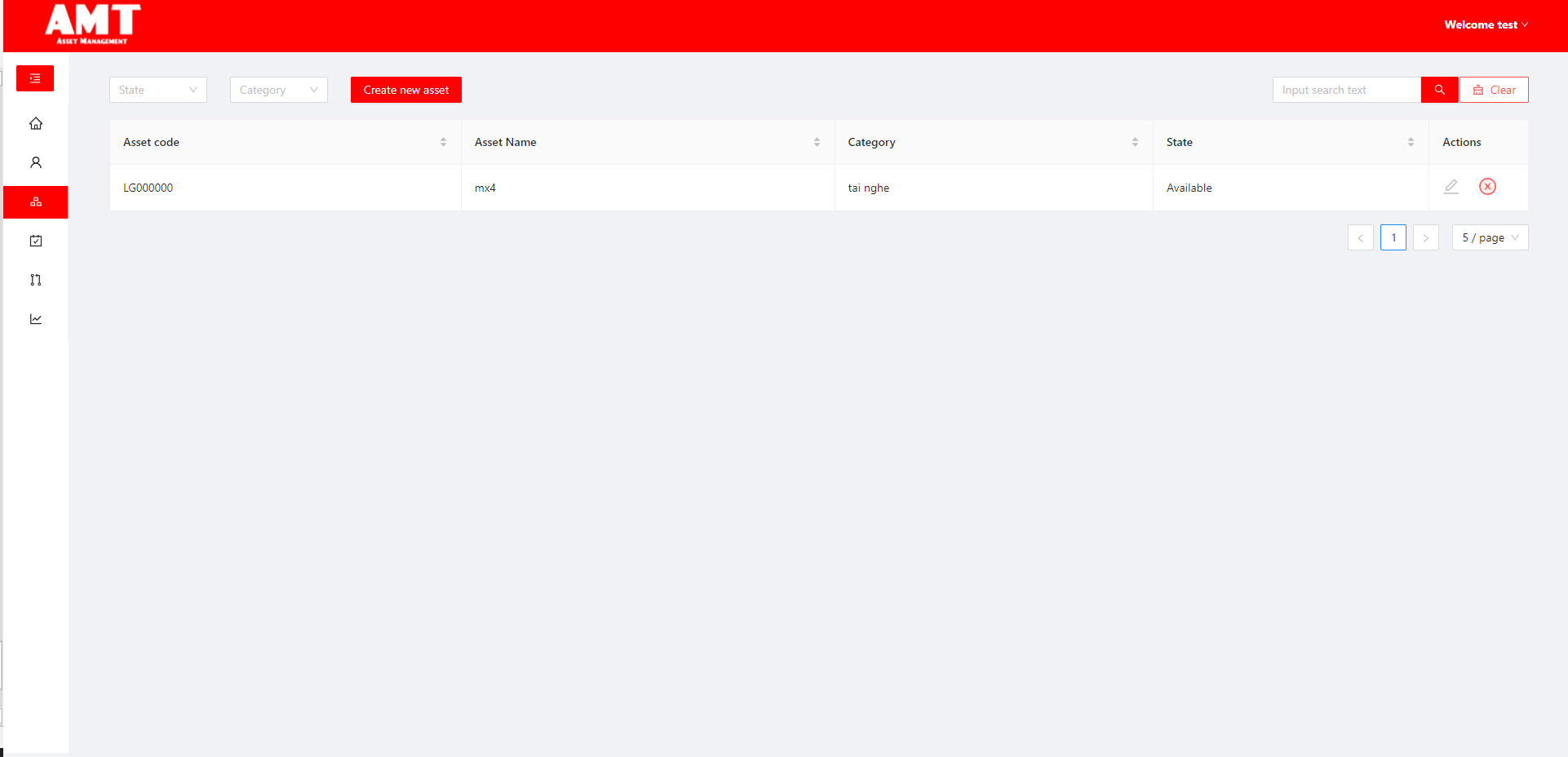
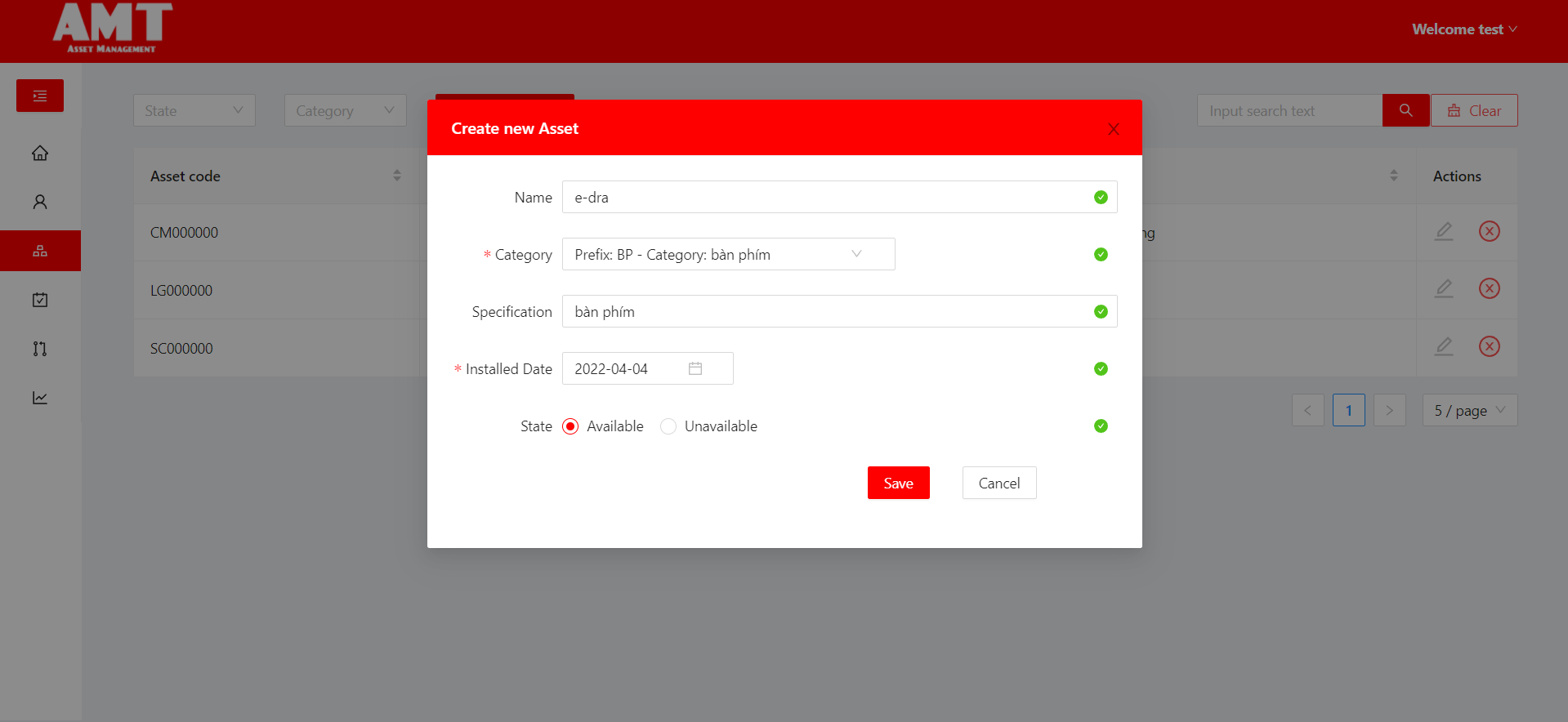


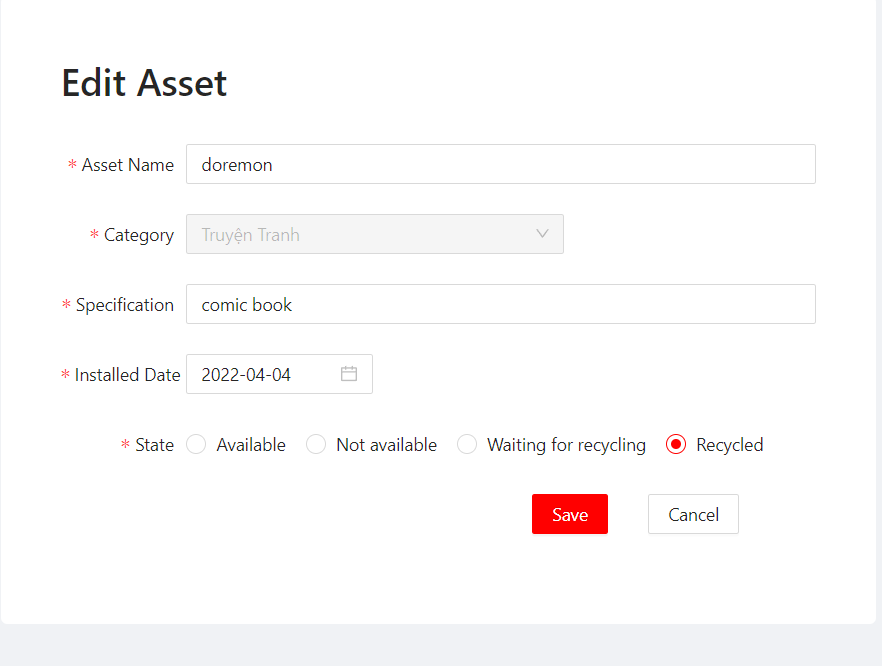
Figure 2 tạo tài sản mới

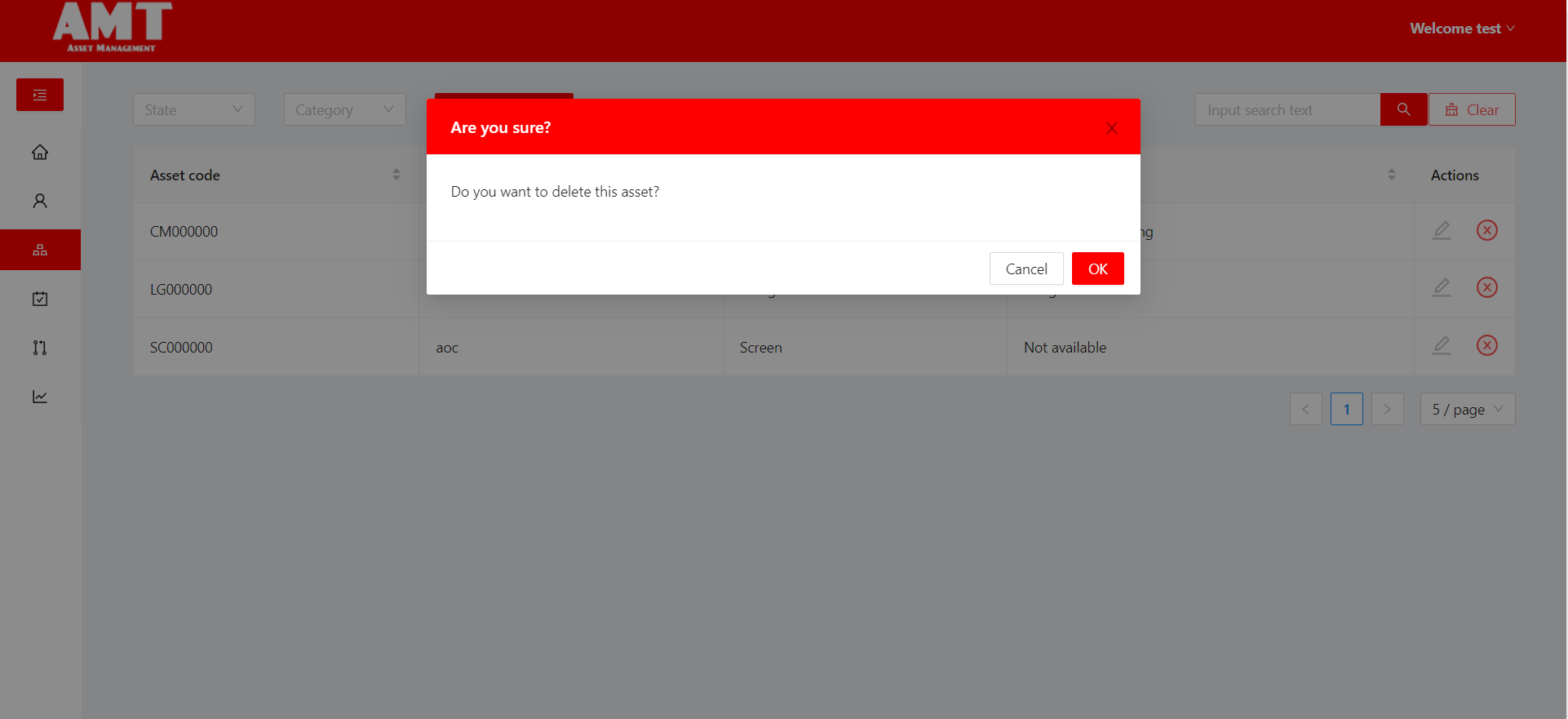
Figure sửa tài khoản

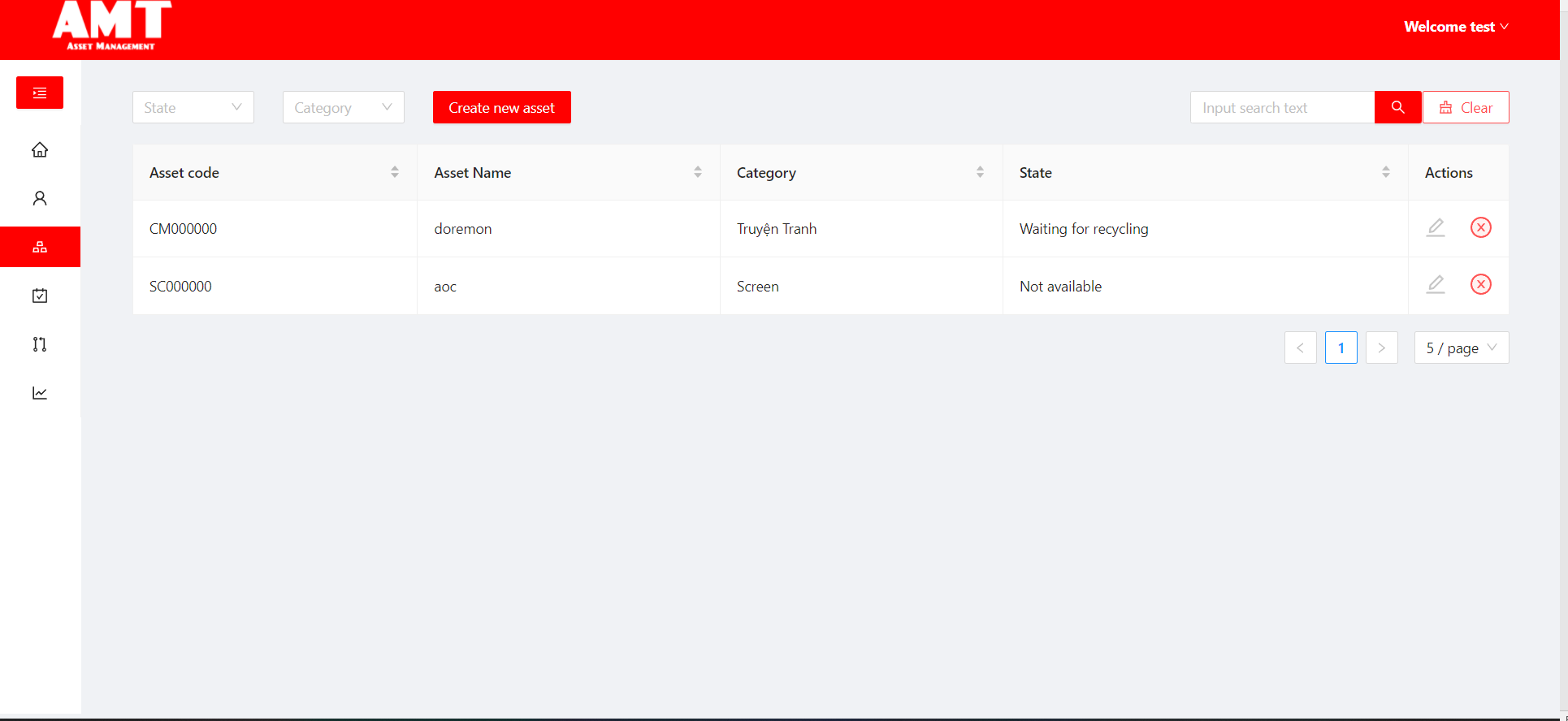
Figure vô hiệu hóa tài khoản

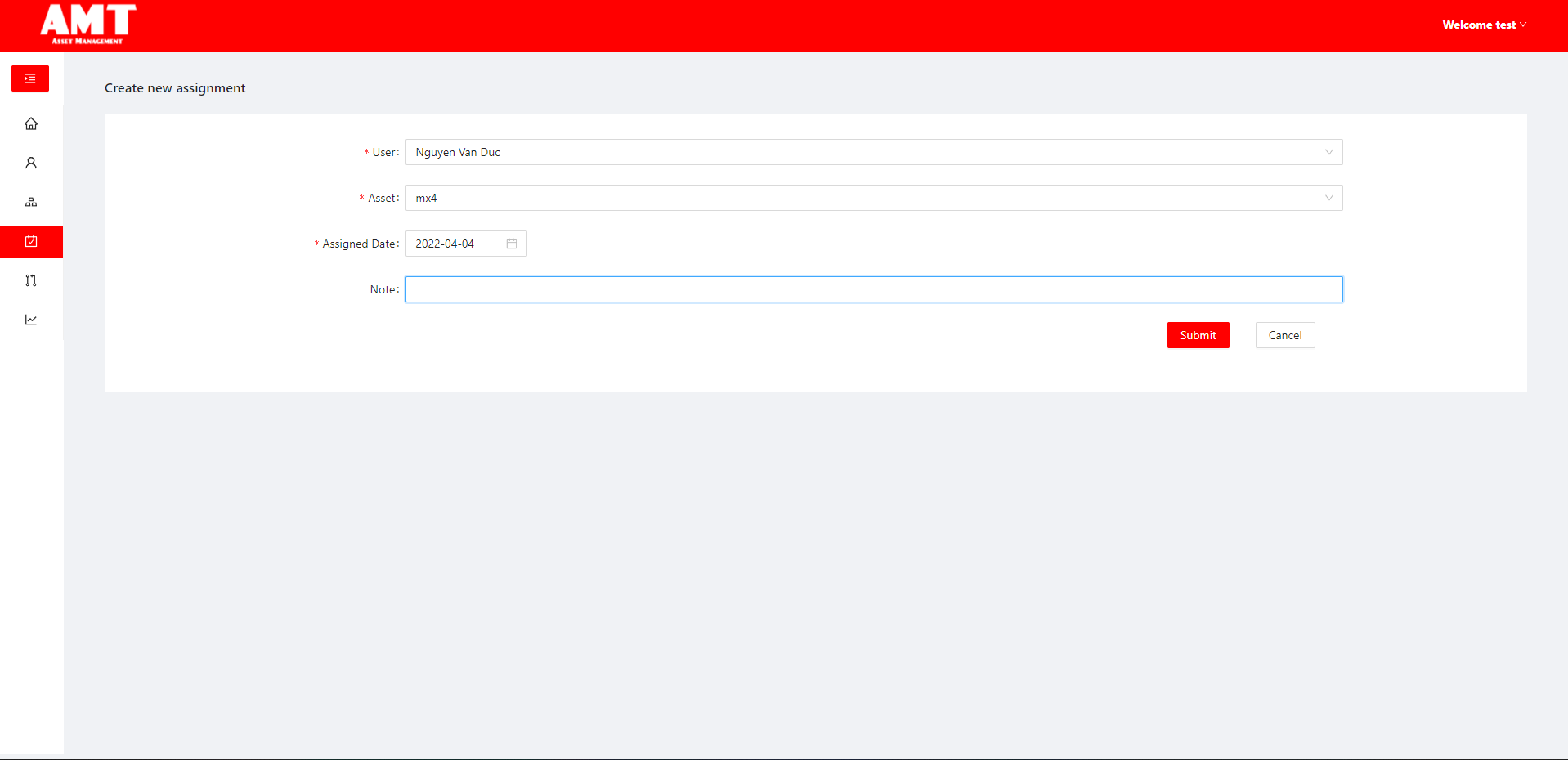
Figure 5 thêm mới thành công

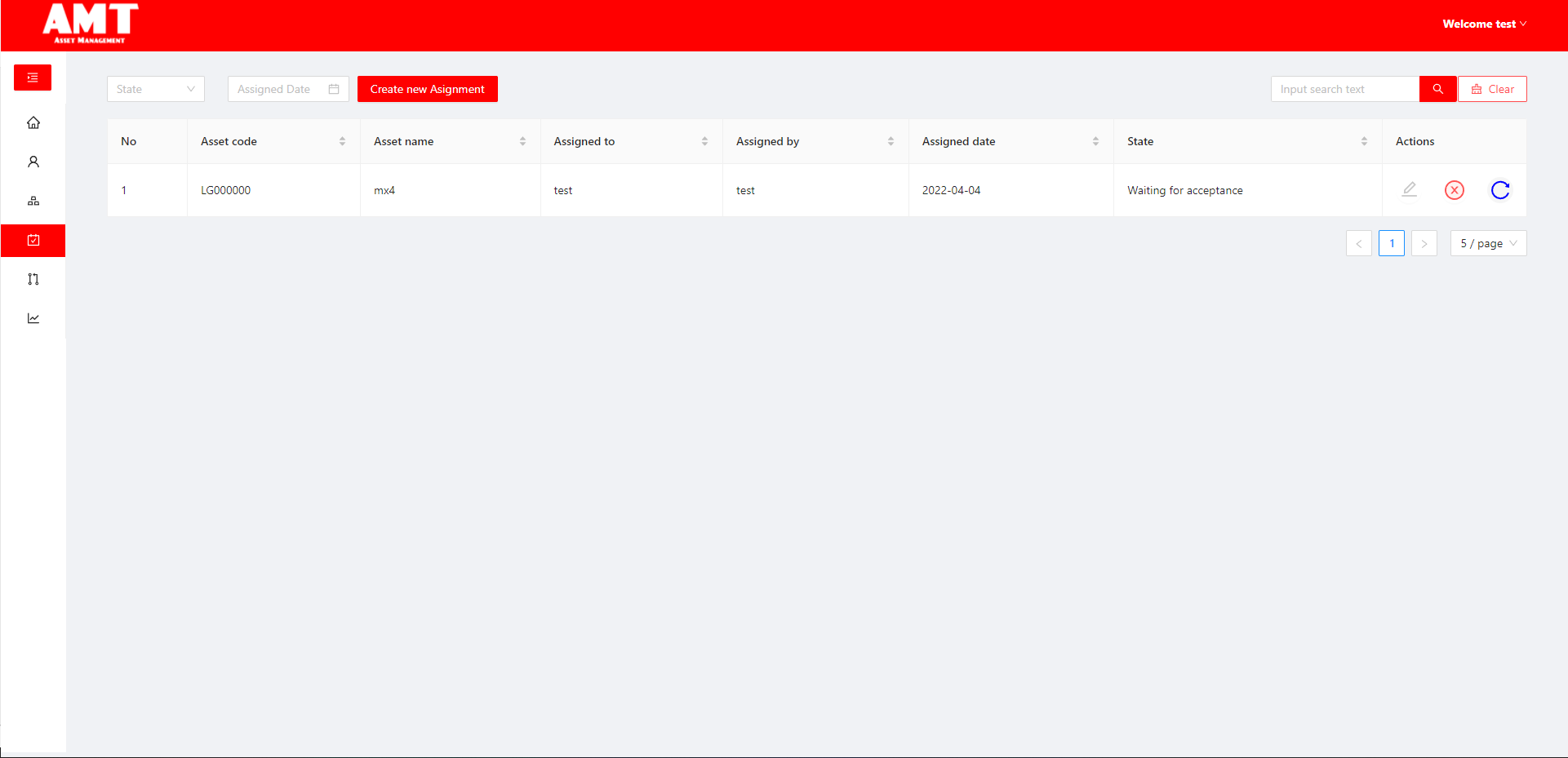
Figure tạo tài sản mới

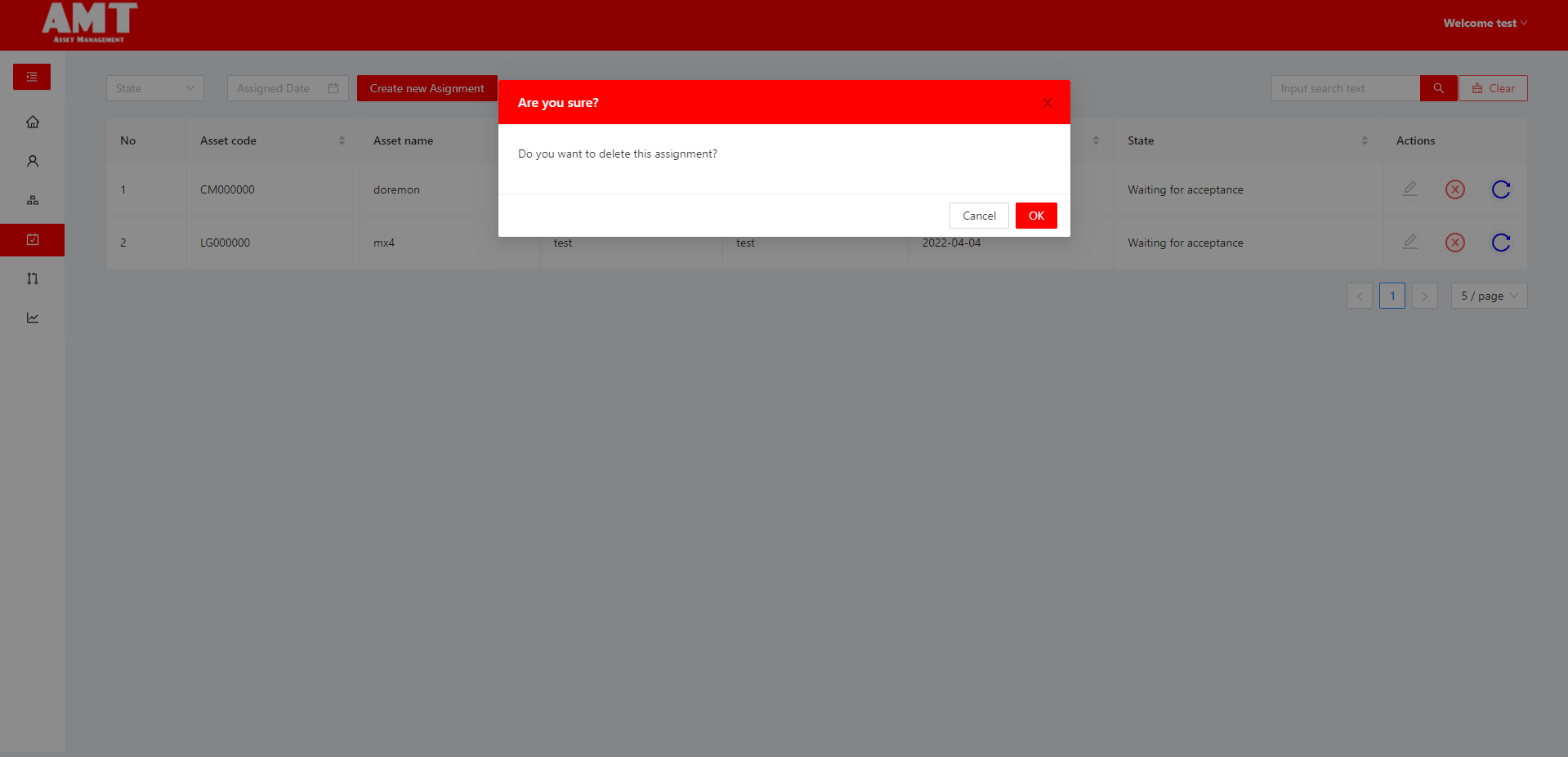
Figure sửa tài sản

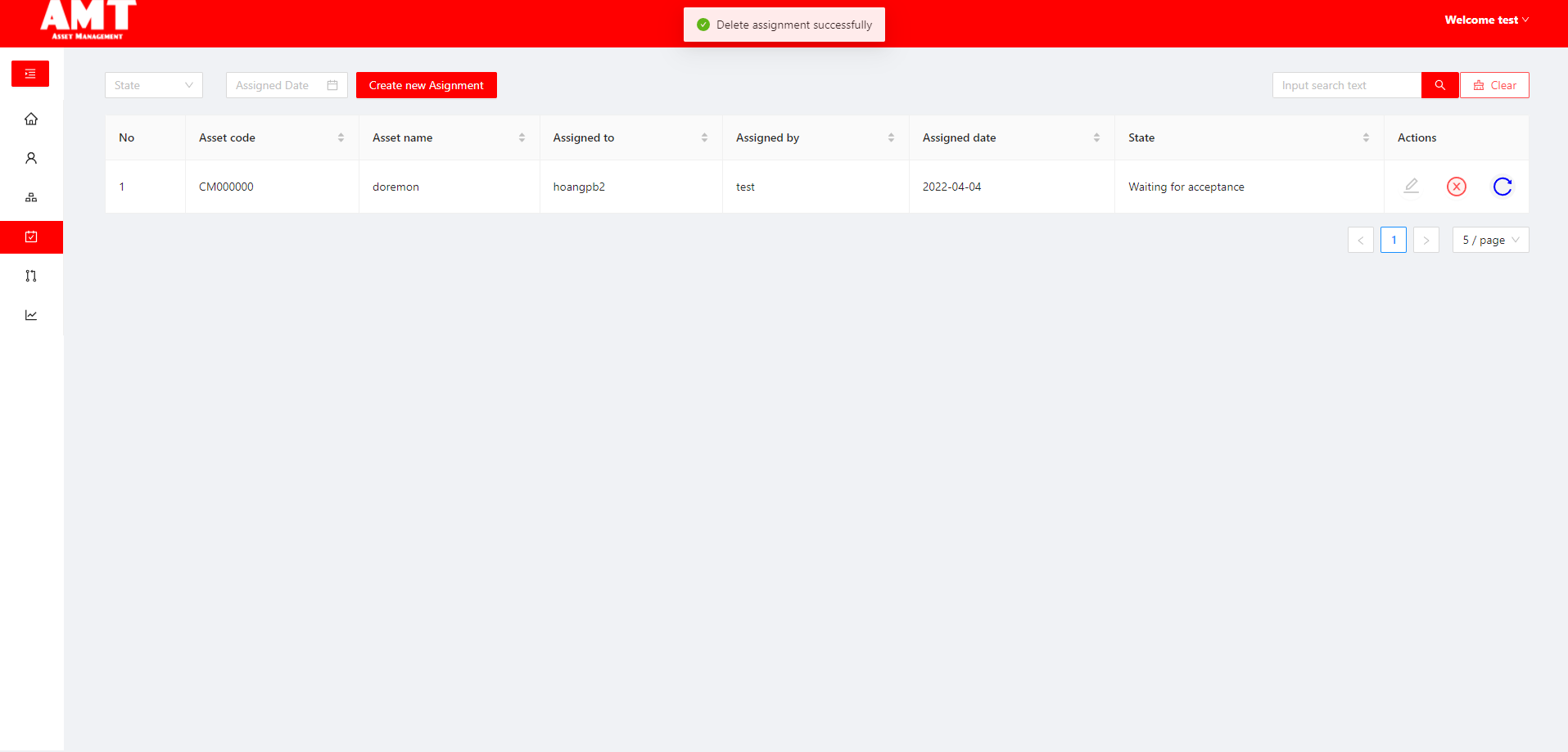
Figure xóa tài sản

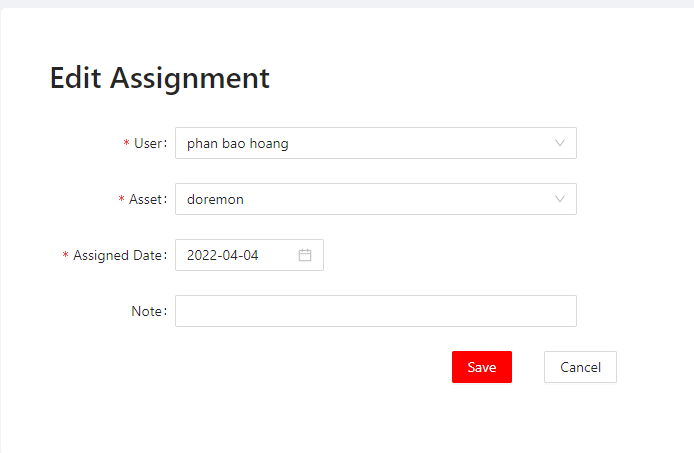
Figure xóa tài sản thành công

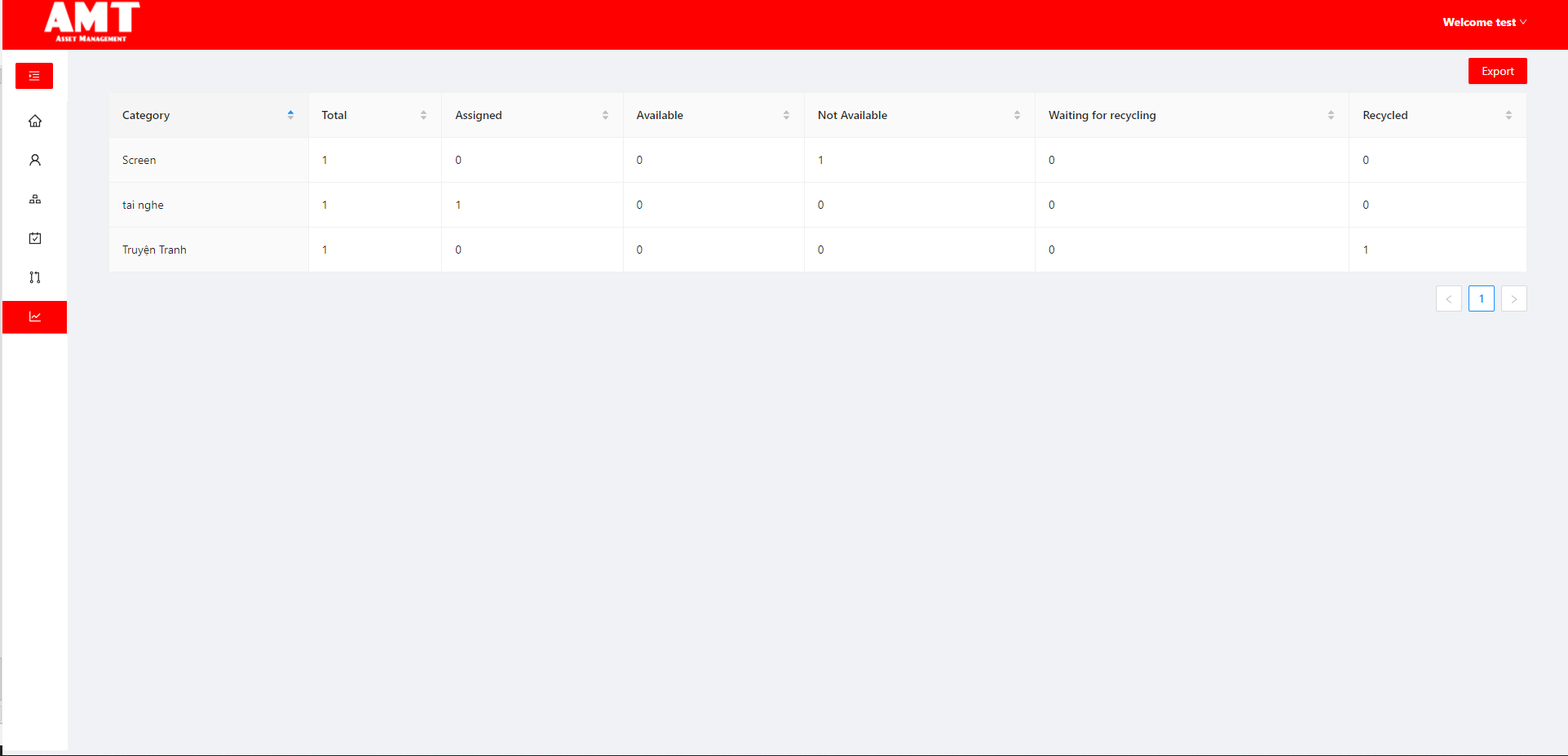
Figure 10 tạo phân công mới

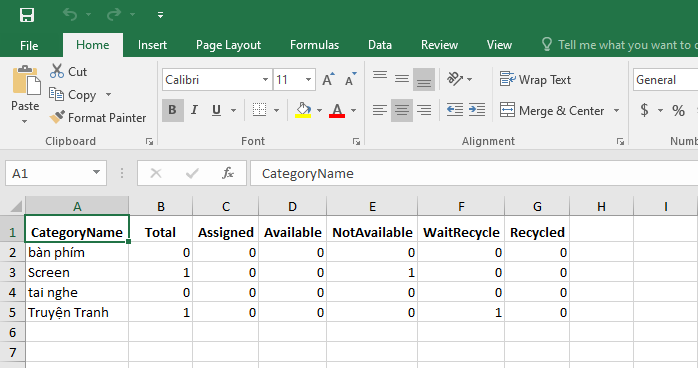
Figure 11 tạo phân công thành công

Figure 12 xóa phân công

Figure 13 xóa phân công thành công

Figure 14 sửa phân công

Figure 15 báo cáo về số lượng, tình trạng

Figure xuất ra file excel