 **BỘ GIÁO DỤC VÀ ĐÀO TẠO  
TRƯỜNG ĐẠI HỌC SƯ PHẠM KĨ THUẬT TP.HCM  
KHOA ĐÀO TẠO CHẤT LƯỢNG CAO**

**Phát triển hệ thống AI Camera Dashboard**

**LỚP: OOSE330679\_22\_2\_01CLC (Sáng thứ bảy)**

**Môn: Công nghệ phần mềm Hướng đối tượng  
NHÓM: 8  
GVHD: TS Huỳnh Xuân Phụng**

***Thành phố Hồ Chí Minh, Tháng năm 2023***

**DANH SÁCH NHÓM**

HỌC KÌ II, NĂM HỌC: 2022-2023

**Tên đề tài: Phát triển hệ thống Camera Dashboard dựa trên thiết kế figma.**

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Họ và tên | MSSV | Mức độ đóng góp |
| 1 | Trần Văn Dân | 20110451 | 100% |
| 2 | Võ Trần Bảo Nguyên | 20110138 | 100% |
| 3 | Huỳnh Hồ Thọ Tỷ | 20110597 | 100% |

**Nhận xét của giáo viên:**

..................................................................................................................................................

..................................................................................................................................................

..................................................................................................................................................

..................................................................................................................................................

*Ngày ........tháng........năm.......*

*Giáo viên chấm điểm*

**Mục lục**

[1. Tổng quan 2](#_Toc134699439)

[1.1. Mô tả hệ thống 2](#_Toc134699440)

[1.2. Phạm vi, đối tượng 2](#_Toc134699441)

[2. Cơ sở các lý thuyết 2](#_Toc134699442)

[2.1. Công nghệ phần mềm hướng đối tượng 2](#_Toc134699443)

[2.2. Các công nghệ, công cụ hỗ trợ 3](#_Toc134699444)

[2.2.1. Back-end: Spring boot 3](#_Toc134699445)

[2.2.2. Front-end: React 4](#_Toc134699446)

[3. Thiết kế hệ thống 5](#_Toc134699447)

[3.1. Thiết kế Class Diagram 5](#_Toc134699448)

[3.2. Thiết kế Usercase 6](#_Toc134699449)

[3.3. Thiết kế Flow Of Event 6](#_Toc134699450)

[3.4. Ràng buộc hệ thống 13](#_Toc134699451)

[4. Thực nghiệm, đánh giá 15](#_Toc134699452)

[4.1. Giao diện các thành phần trang của phần mềm 15](#_Toc134699453)

[4.2. Hướng dẫn sử dụng 18](#_Toc134699454)

[5. Kết luận 24](#_Toc134699455)

[5.1. Kết quả đạt được 24](#_Toc134699456)

[5.2. Định hướng phát triển 24](#_Toc134699457)

# Tổng quan

## Mô tả hệ thống

Hệ thống giám sát bằng Camera được dùng để giám sát các khu vực trong công ty, thực hiện việc gửi cảnh báo khi có người lạ hoặc có Nhân viên không phận sự xuất hiện trong khu vực. Mỗi Nhân viên của công ty sẽ được lưu trữ các thông tin nhận dạng như tên, số điện thoại, địa chỉ, thông tin nhận dạng (Khuôn mặt,...), thông tin các khu vực bị cấm xuất hiện, chỉ duy nhất các quản lý mới có tài khoản đăng nhập vào hệ thống quản lý. Công ty sẽ được chia thành nhiều khu vực, mỗi khu vực chỉ có 1 camera. Mỗi Camera sẽ được tích hợp hệ thống AI để hỗ trợ nhận diện phát hiện khuôn mặt. Khi có bất kì sự xuất hiện bất thường nào xảy ra ở các khu vực chỉ định, hệ thống sẽ gửi một Thông tin Cảnh báo, Mỗi Thông tin Cảnh Báo sẽ lưu các thông tin ngày giờ phát cảnh báo, video cảnh báo và sẽ thuộc duy nhất 1 loại Cảnh báo. Cảnh báo sẽ được tạo do quản lý để Khu vực, Mỗi alert chỉ thuộc 1 Khu Vực. Có 2 loại cảnh báo, cảnh báo người lạ và cảnh báo có nhân viên bị cấm xuất hiện trong khu vực.

## Phạm vi, đối tượng

- Phạm vi: Lý thuyết thiết kế phần mềm theo hướng đối tượng, các quy trình làm việc, kỹ thuật lập trình phần mềm, sử dụng các Framework của javascript và java, kỹ thuật deploy lên AWS, Docker.

- Đối tượng: phần mềm dùng để quản lý các camera gắn trong các khu vực chỉ định, các camera có thể phát hiện và gửi cảnh báo theo các điều kiện đã đề ra.

# Cơ sở các lý thuyết

## Công nghệ phần mềm hướng đối tượng

***Các bước phát triển một phần mềm theo công nghệ hướng đối tượng:***

- Phân tích yêu cầu: Hiểu và xác định các yêu cầu chức năng và phi chức năng của phần mềm. Tạo một bản mô tả chi tiết về các tính năng và khả năng của hệ thống. Kết quả của bước này sẽ cho ra tài liệu SRS.

- Thiết kế hệ thống: Xây dựng kiến trúc tổng thể của hệ thống. Chia nhỏ hệ thống thành các đối tượng, xác định mối quan hệ giữa chúng và xác định các phương thức và thuộc tính của mỗi đối tượng. Tạo ra các biểu đồ lớp, biểu đồ tuần tự và các biểu đồ khác để trình bày thiết kế.

- Triển khai lớp: Xây dựng các lớp và các đối tượng cụ thể dựa trên thiết kế đã hoàn chỉnh. Viết mã và triển khai các lớp và các phương thức của chúng.

- Kiểm thử và gỡ lỗi: Thực hiện các bài kiểm tra đơn vị và kiểm thử hộp đen và hộp trắng để đảm bảo tính đúng đắn và hoạt động chính xác của hệ thống. Sửa lỗi và tối ưu hóa hiệu suất.

- Điều chỉnh và tối ưu hóa: Kiểm tra, đánh giá và tối ưu hóa các thành phần của hệ thống để đảm bảo hiệu suất tốt nhất và tuân thủ các tiêu chuẩn và hướng dẫn.

- Triển khai: Đóng gói và triển khai phần mềm. Cài đặt hệ thống vào môi trường sản xuất và thực hiện kiểm tra cuối cùng để đảm bảo tính ổn định và hoạt động như mong đợi.

- Bảo trì và phát triển: Theo dõi và duy trì phần mềm trong suốt quá trình vận hành. Thực hiện các bản vá lỗi, nâng cấp và phát triển thêm tính năng theo yêu cầu mới của người dùng.

## Các công nghệ, công cụ hỗ trợ

### Back-end: Spring boot

Spring Boot là một framework phát triển ứng dụng Java dựa trên Spring Framework. Nó giúp cho việc xây dựng ứng dụng Java trở nên nhanh chóng và dễ dàng hơn bằng cách cung cấp một cách tiếp cận tối giản hóa việc cấu hình và triển khai ứng dụng.

Spring Boot có một số đặc điểm quan trọng. Đầu tiên, nó cung cấp một tập hợp các dependency tự động, giúp giảm thời gian và công sức cấu hình. Nó cũng có thể nhúng máy chủ ứng dụng như Tomcat hoặc Undertow trực tiếp vào ứng dụng, loại bỏ nhu cầu phải triển khai riêng biệt. Spring Boot cũng cung cấp một bộ công cụ phát triển hỗ trợ hot-reloading, giúp phát triển ứng dụng một cách nhanh chóng và tiện lợi.

Một điểm mạnh khác của Spring Boot là khả năng tích hợp tốt với các công nghệ và thành phần khác nhau. Nó hỗ trợ các công nghệ như Spring Data, Spring Security, và Spring Integration, cho phép xây dựng các ứng dụng phức tạp và an toàn. Ngoài ra, Spring Boot cũng cung cấp các công cụ quản lý ứng dụng như Spring Actuator, giúp theo dõi và quản lý ứng dụng một cách dễ dàng. (Ref: Wikipedia)

### Front-end: React

React là một thư viện JavaScript phổ biến được sử dụng để xây dựng giao diện người dùng cho các ứng dụng web. Được phát triển bởi Facebook, React tập trung vào việc tạo ra các thành phần UI tái sử dụng và dễ quản lý.

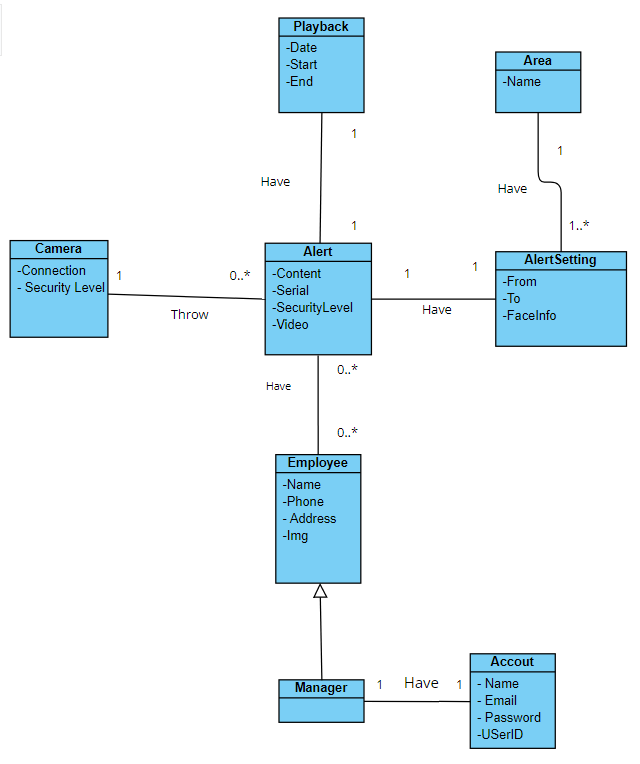
React sử dụng một khái niệm gọi là "Virtual DOM" để tăng hiệu suất và tối ưu hóa việc cập nhật giao diện. Thay vì cập nhật toàn bộ giao diện khi có thay đổi, React sẽ chỉ cập nhật các thành phần có thay đổi, giúp giảm tải cho trình duyệt và cải thiện trải nghiệm người dùng.

Một điểm mạnh của React là khả năng tái sử dụng thành phần. Thành phần React có thể được tạo ra một lần và sử dụng lại ở nhiều nơi trong ứng dụng, giúp tạo ra mã dễ bảo trì và mở rộng. React cũng hỗ trợ viết mã trong JSX, một cú pháp kết hợp giữa JavaScript và HTML, giúp xây dựng giao diện trực quan và dễ đọc.

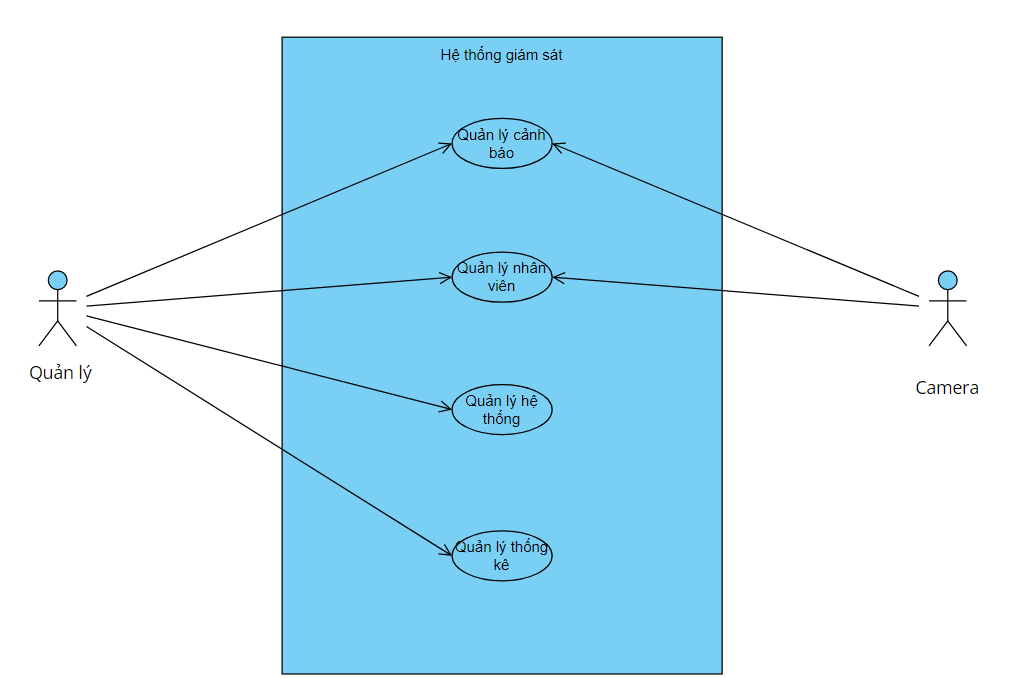
Ngoài ra, React cũng được hỗ trợ bởi một cộng đồng lớn và phong phú, với nhiều công cụ và thư viện phụ trợ. Các công nghệ bổ sung như React Router và Redux cung cấp giải pháp cho việc quản lý định tuyến và trạng thái ứng dụng. (Ref: Wikipedia)

# Thiết kế hệ thống

## Thiết kế Class Diagram



## Thiết kế Usercase



## Thiết kế Flow Of Event

|  |  |
| --- | --- |
| Event | Gửi cảnh báo |
| Actor | Camera |
| Precondition | Có người đi vào khu vực cấm có camera đang quản lý  Hệ thống đã cài đặt khu vực cấm và danh sách nhân viên bị cấm cho camera |
| Postcondition | Gửi cảnh báo khi có người vào khu vực cấm |
| Basic Flow | 1. Có người đi vào khu vực bị cấm có camera giám sát 2. Bắt đầu ghi lại video 3. Thực hiện subflow: Kiếm tra thông tin khuôn mặt 4. Ghi nhận dữ liệu nhân viên bị cấm (If không có dữ liệu được ghi nhận) A1: Thực hiện Alternative flow: Xác định người lạ 5. Gửi cảnh báo tới hệ thống |
| Sub flow | S1: Kiểm tra thông tin khuôn mặt |
|  | 1. Lấy dữ liệu khuôn mặt của người lạ 2. So sánh dữ liệu vừa lấy với dữ liệu của các nhân viên bị cấm |
| Alternative flow | A1: Xác định người lạ |
|  | 1. Chụp lại hình ảnh khuôn mặt của người lạ 2. Ghi nhận thông tin 3. Gửi cảnh báo tới hệ thống |

|  |  |
| --- | --- |
| Event | Xem cảnh báo |
| Actor | Quản lý |
| Precondition | Cảnh báo xuất hiện, được gửi từ Camera |
| Postcondition | Xem được các thông tin như video, thời gian, khu vực, camera nào gửi cảnh báo |
| Basic Flow | 1. Đăng nhập hệ thống  Nhập tài khoản và mật khẩu  Nếu nhập sai, thực hiện Alternative flow: Nhập chưa đúng  2. Mở tab “Cảnh báo”  3. Mở Cảnh báo muốn xem |
| Alternative flow | A1: Nhập chưa đúng |
|  | 1. Gửi thông báo nhập sai  2. Nhập lại tài khoản mật khẩu  Nếu tiếp tục sai, thực hiện lại Alternative flow: Nhập chưa đúng  3. Mở tab Cảnh báo  4. Mở cảnh báo muốn xem |

|  |  |
| --- | --- |
| Event | Tạo profile nhân viên |
| Actor | Quản lý |
| Precondition | Quản lý đã đăng nhập vào hệ thống |
| Postcondition | Tạo được một profile mới |
| Basic Flow | 1. Chọn chức năng thêm profile 2. Nhập thông tin nhân viên 3. Kiểm tra nhân viên đã có trong hệ thống hay chưa 4. Xác nhận tạo   If nhân viên đã có trong hệ thống, thực hiện Altinative flow: Thông báo tạo thất bại   1. Thông báo tạo thành công   Gửi thông báo đã tạo thành công profile |
| Alternative flow | A1: Thông báo tạo thất bại |
|  | 1. Gửi thông báo tạo profile thất bại do nhân viên đã tồn tại |

|  |  |
| --- | --- |
| Event | Xem profile nhân viên |
| Actor | Quản lý |
| Precondition | Quản lý đã đăng nhập vào hệ thống |
| Postcondition | Xem được thông tin nhân viên |
| Basic Flow | 1. Chọn chức năng xem thông tin nhân viên 2. Hiển thị thông tin nhân viên |

|  |  |
| --- | --- |
| Event | Tạo báo cáo từ thống kê |
| Actor | Quản lý |
| Precondition | Quản lý đã đăng nhập vào hệ thống |
| Postcondition | Tạo được một báo cáo mới |
| Basic Flow | 1. Chọn tab thống kê 2. Chọn chức năng tạo báo cáo 3. Hệ thống thông báo tạo thành công 4. Hiển thị báo cáo   If không có thống kê nào: thực hiên Alternative flow: Thông báo tạo thất bại |
| Alternative flow | A1: Thông báo tạo thất bại |
|  | 1. Hiển thị thông báo không thể tạo báo cáo do không có thống kê nào |

|  |  |
| --- | --- |
| Event | Xem thống kê |
| Actor | Quản lý |
| Precondition | Quản lý đã đăng nhập vào hệ thống |
| Postcondition | Hiển thị các thống kê |
| Basic Flow | 1. Chọn tab thống kê 2. Chọn chức năng xem thống kê 3. Hiển thị các thống kê   If không có thống kê nào: thực hiện alternative flow: Thông báo không có thống kê |
| Alternative flow | A1: Thông báo không có thống kê |
|  | 1. Hiển thị thông báo không có thống kê nào tồn tại |

|  |  |
| --- | --- |
| Event | Cài đặt hệ thống |
| Actor | Quản lý |
| Precondition | Hệ thống đã được lắp đặt các camera, quản lý đã đăng nhập vào hệ thống |
| Postcondition | Thay đổi được các thông số như số lượng camera hiển thị, đổi tên camera |
| Basic Flow | 1. Chọn số lượng camera để hiển thị 2. Thay đổi số lượng camera trên màn hình 3. Chọn camera muốn đổi tên 4. Nhập tên cần đổi 5. Bấm xác nhận   5.1 Kiểm tra tên đã tồn tại hay chưa  5.2 If tên camera bị trùng: thực hiện Sub flow: Nhập tên camera.   1. Thông báo đổi tên camera thành công |
| Sub flow | S1: Nhập tên camera |
|  | 1. Nhập tên camera 2. Quay lại bước 5 của Basic Flow |

## Ràng buộc hệ thống

- Chỉ có quản lý mới có thể đăng nhập vào hệ thống.

- Sau khi phát hiện được người lạ hoặc nhân viên vào khu vực không phận sự phải cảnh báo trong vòng 5 giây.

- Thông tin về nhân viên phải có ảnh khuôn mặt.

- Camera phải được kết nối mạng.

- Mỗi camera chỉ được xác định đúng 1 khu vực.

- Camera phải được training bằng hình ảnh nhân viên trước khi đưa vào hoạt động.

- Trước khi xuất báo cáo thống kê, quản lý phải xác nhận lại các cảnh báo(false positve hay không)

- Khi cảnh báo bắt đầu mới bắt đầu tạo Playback

- Thời gian lưu cảnh báo là 3 tháng, quá thời gian sẽ tự xoá.

# Thực nghiệm, đánh giá

## Giao diện các thành phần trang của phần mềm

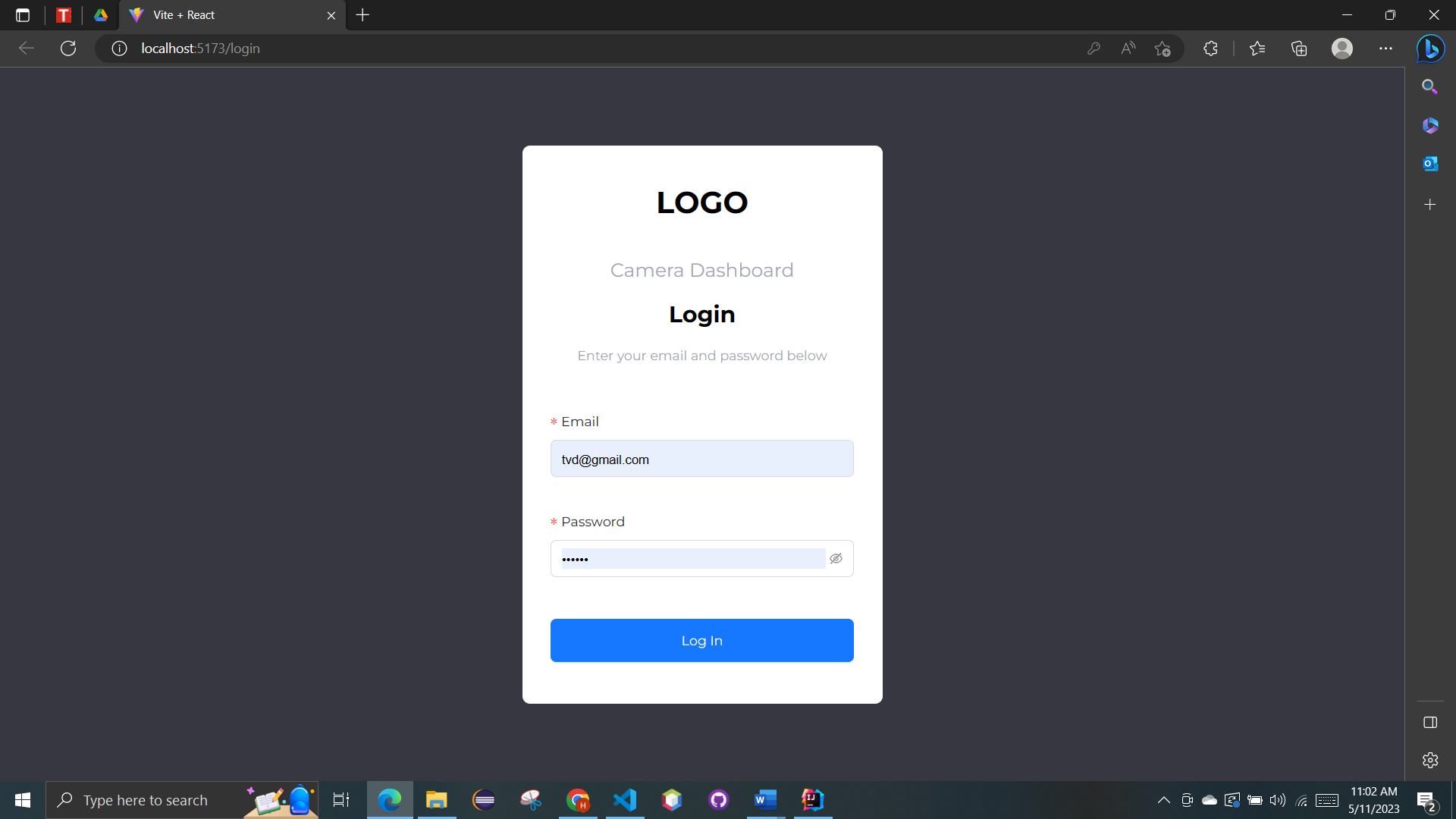


Figure 2.1.Trang login

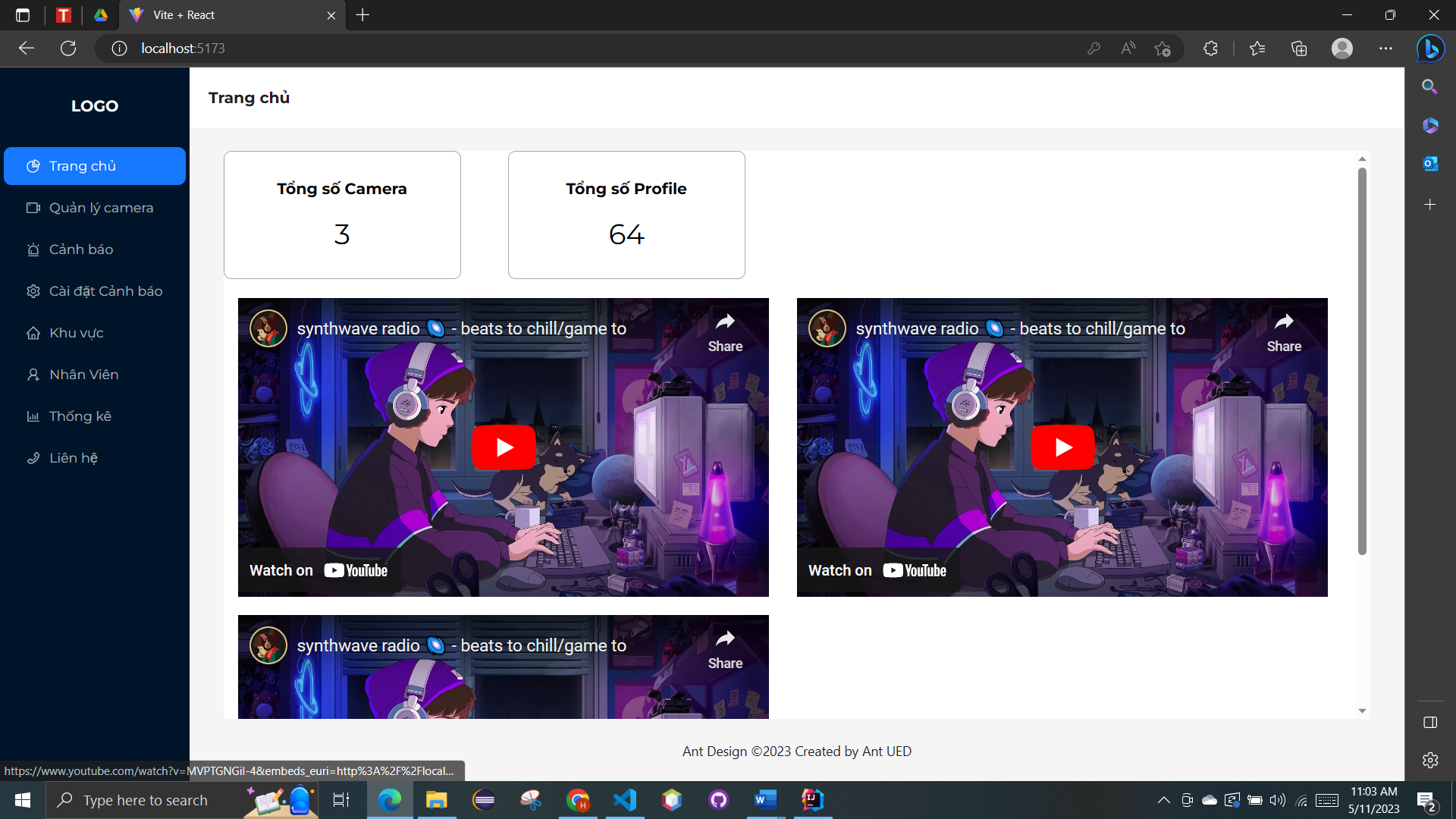


Figure 2.2.Trang chủ

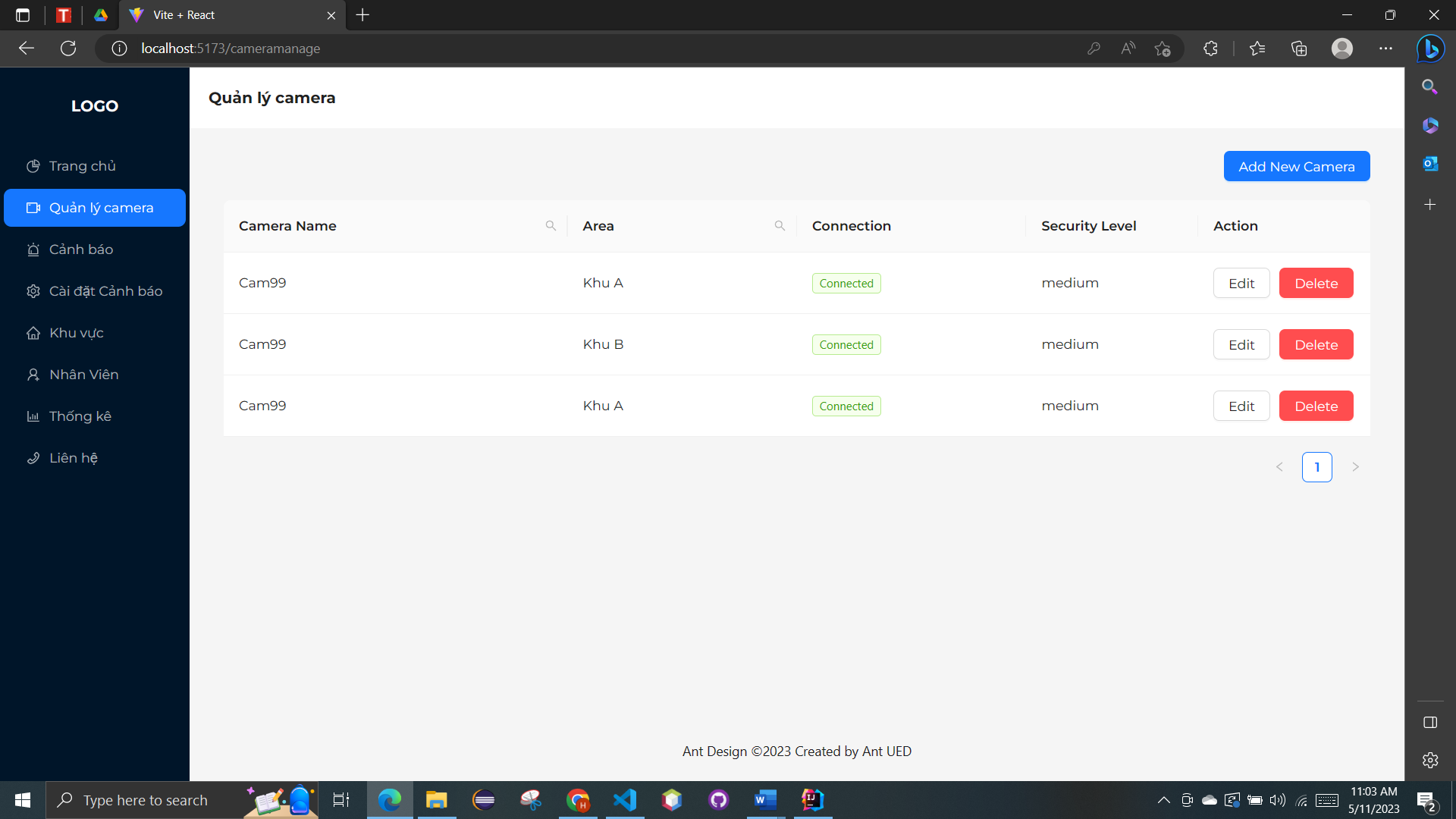


Figure 2.3.Trang quản lý camera

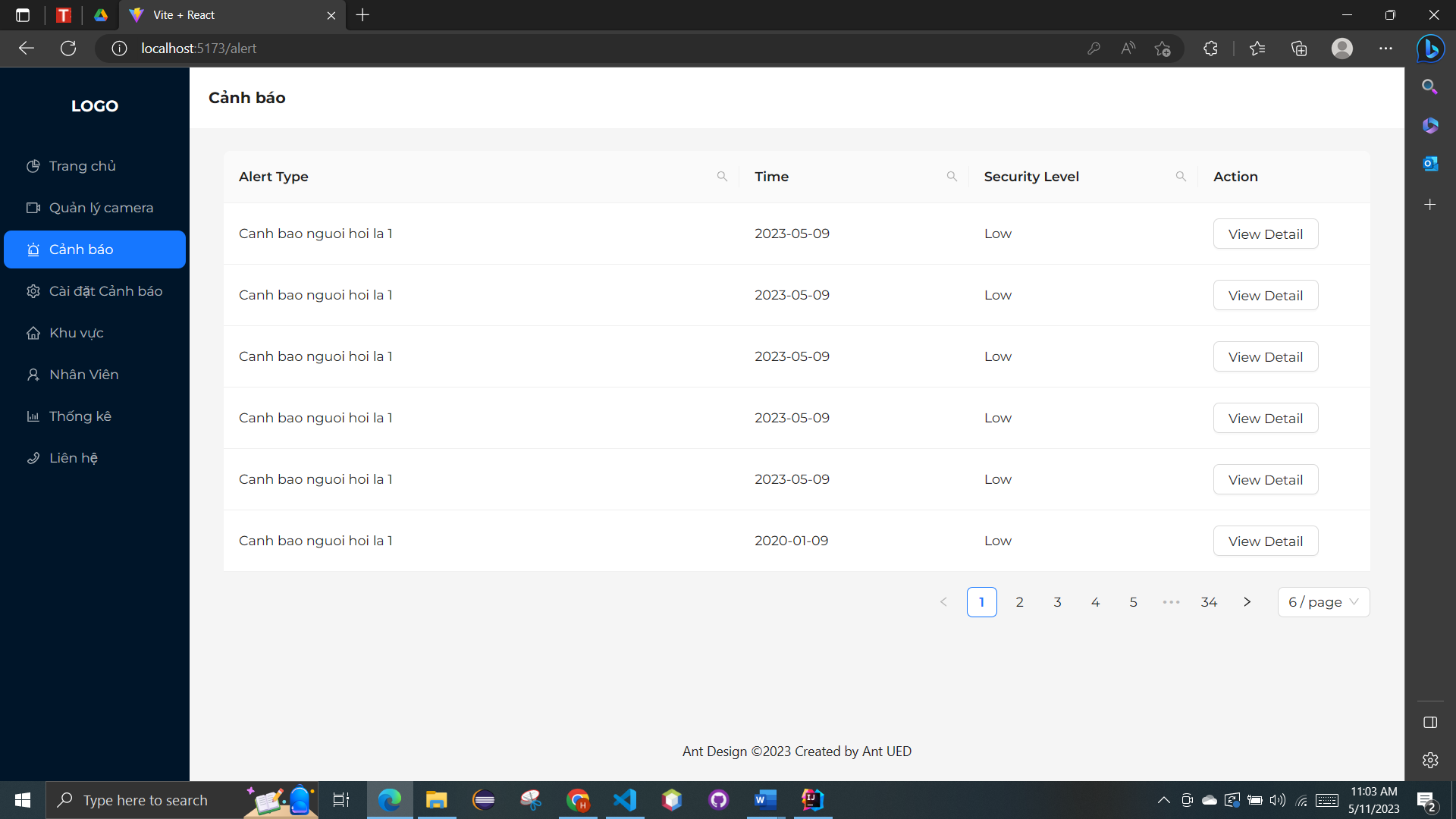


Figure 2.4.Trang quản lý các Cảnh báo

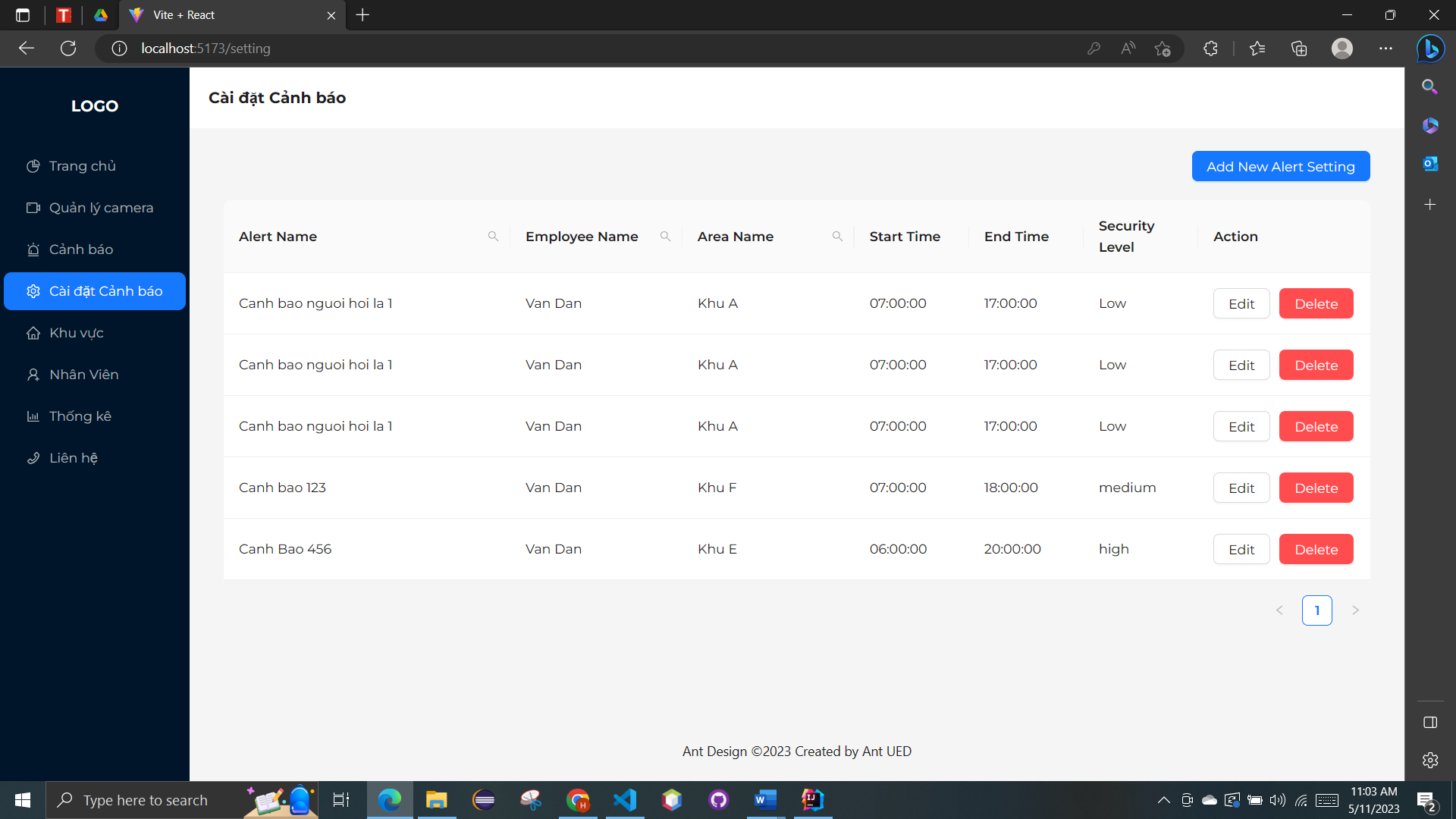


Figure 2.5.Trang cài đặt cảnh báo

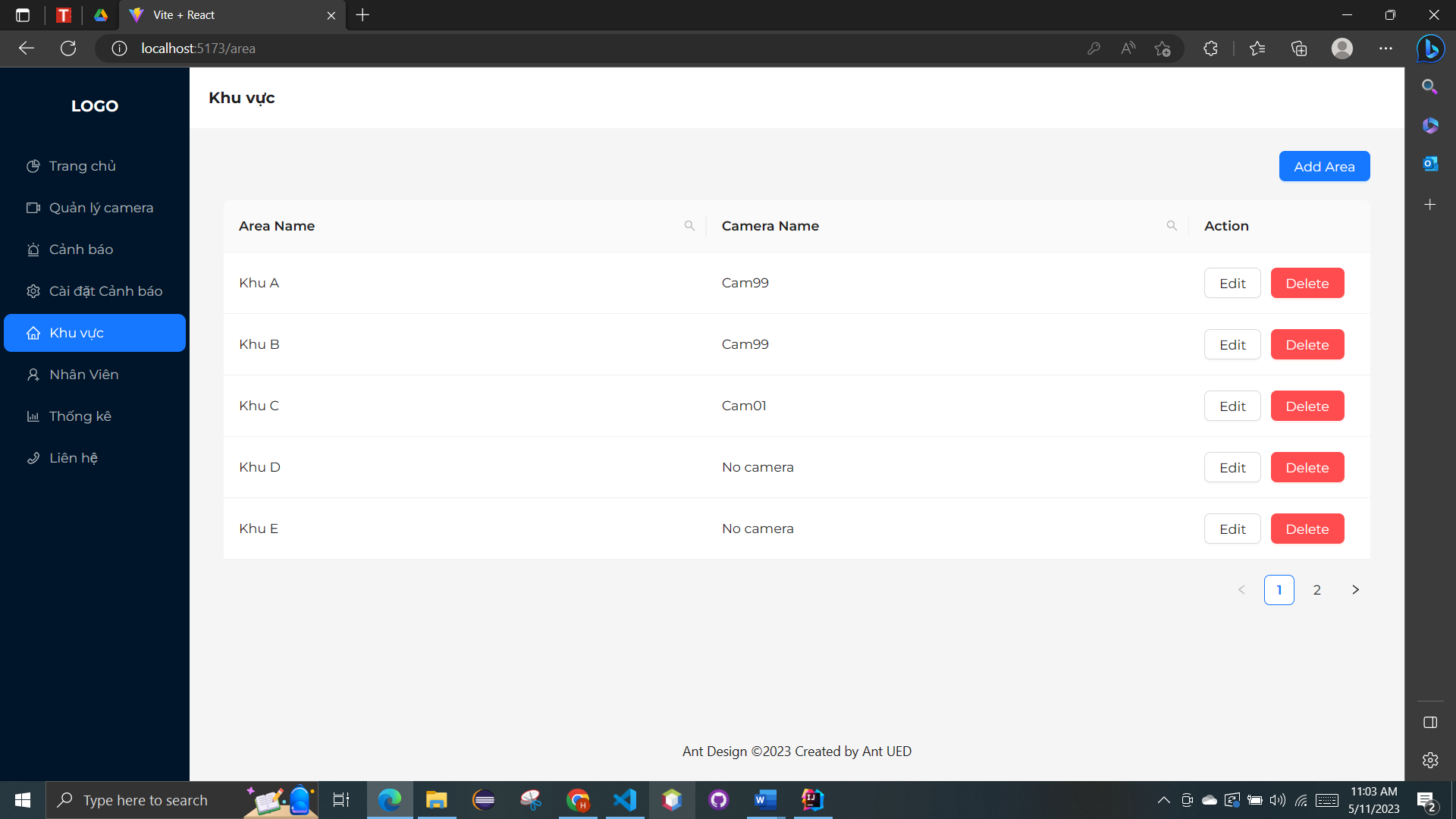


Figure 2.6.Trang quản lý khu vực

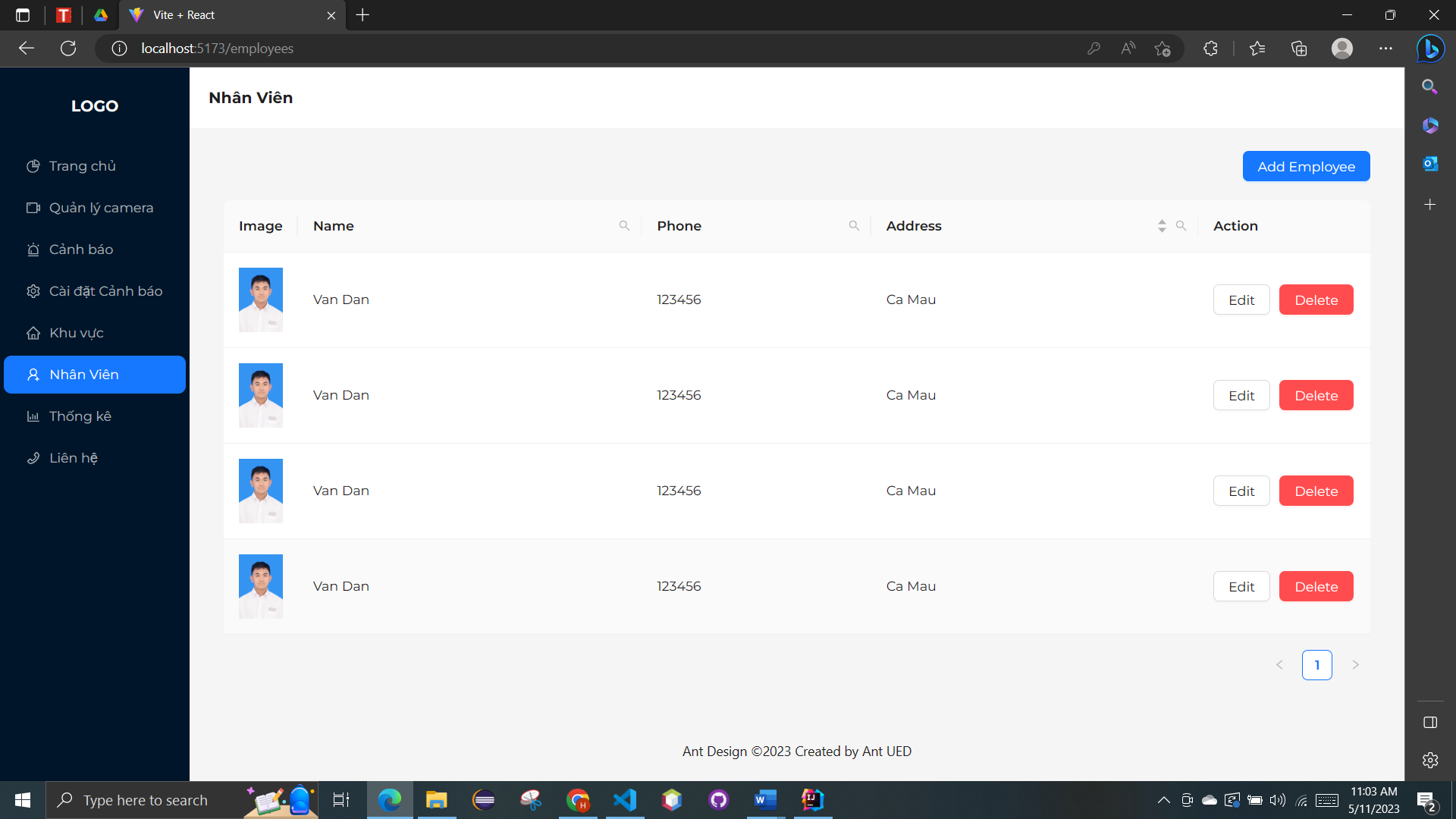


Figure 2.7.Trang quản lý nhân viên

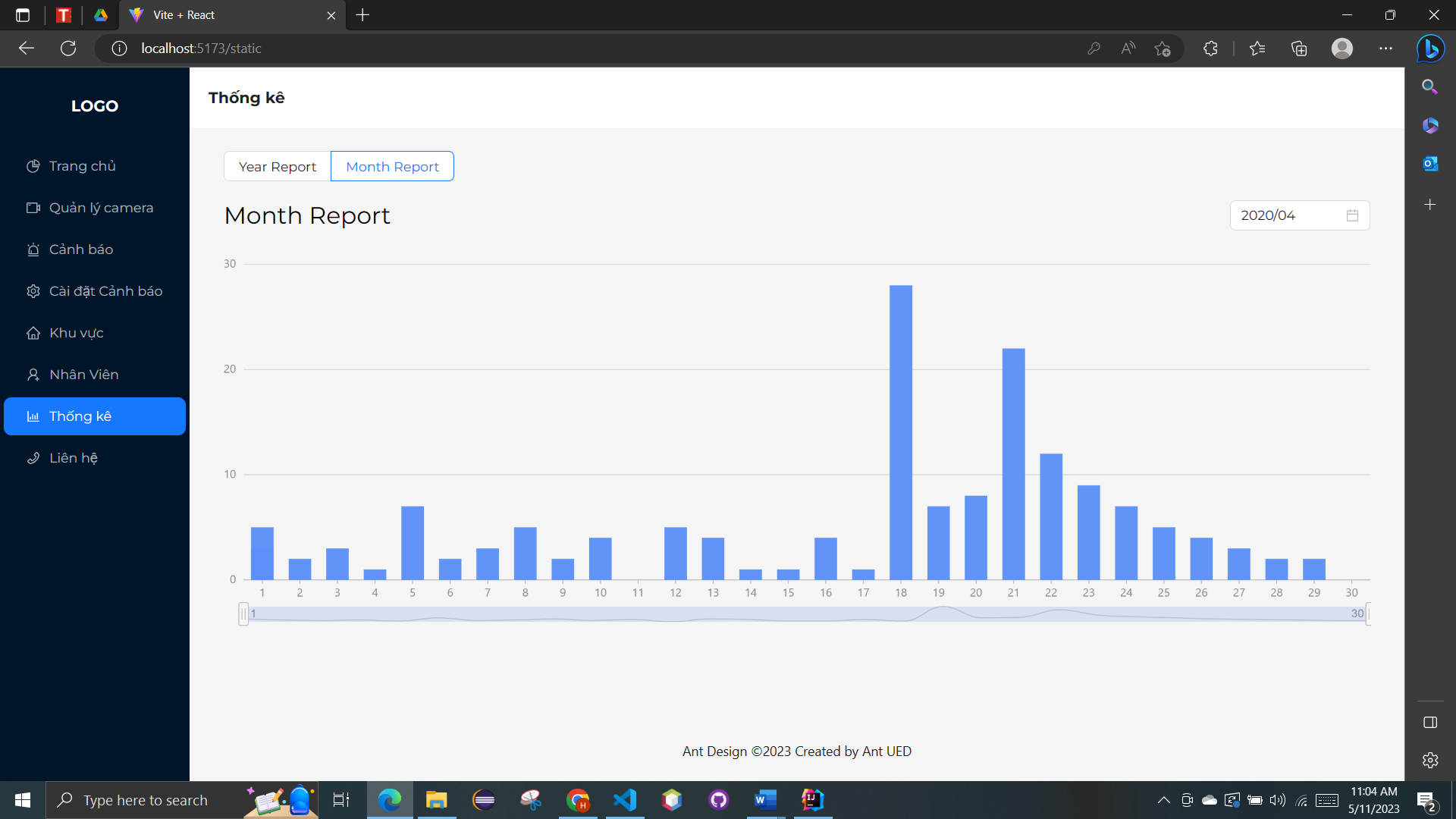
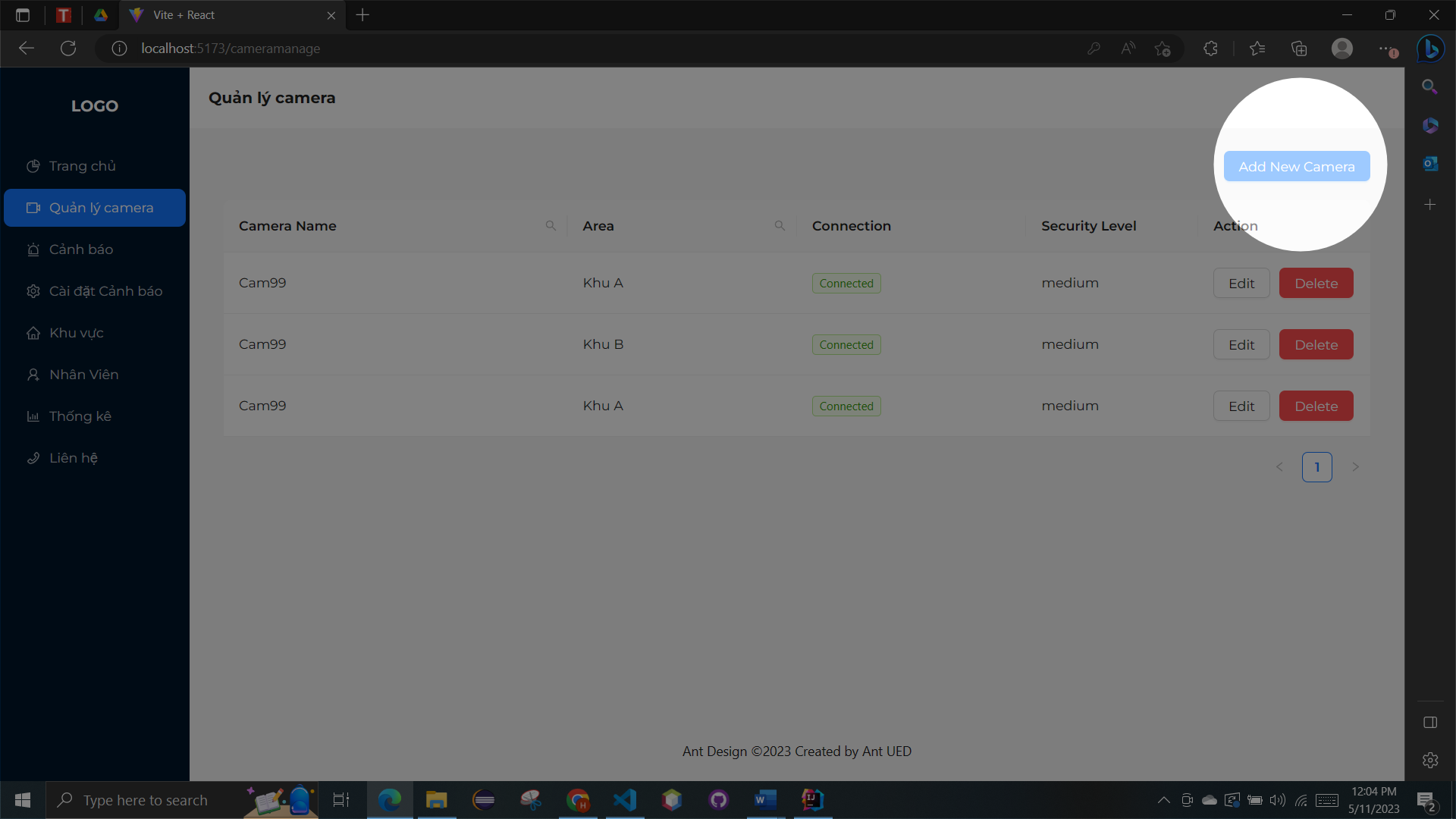


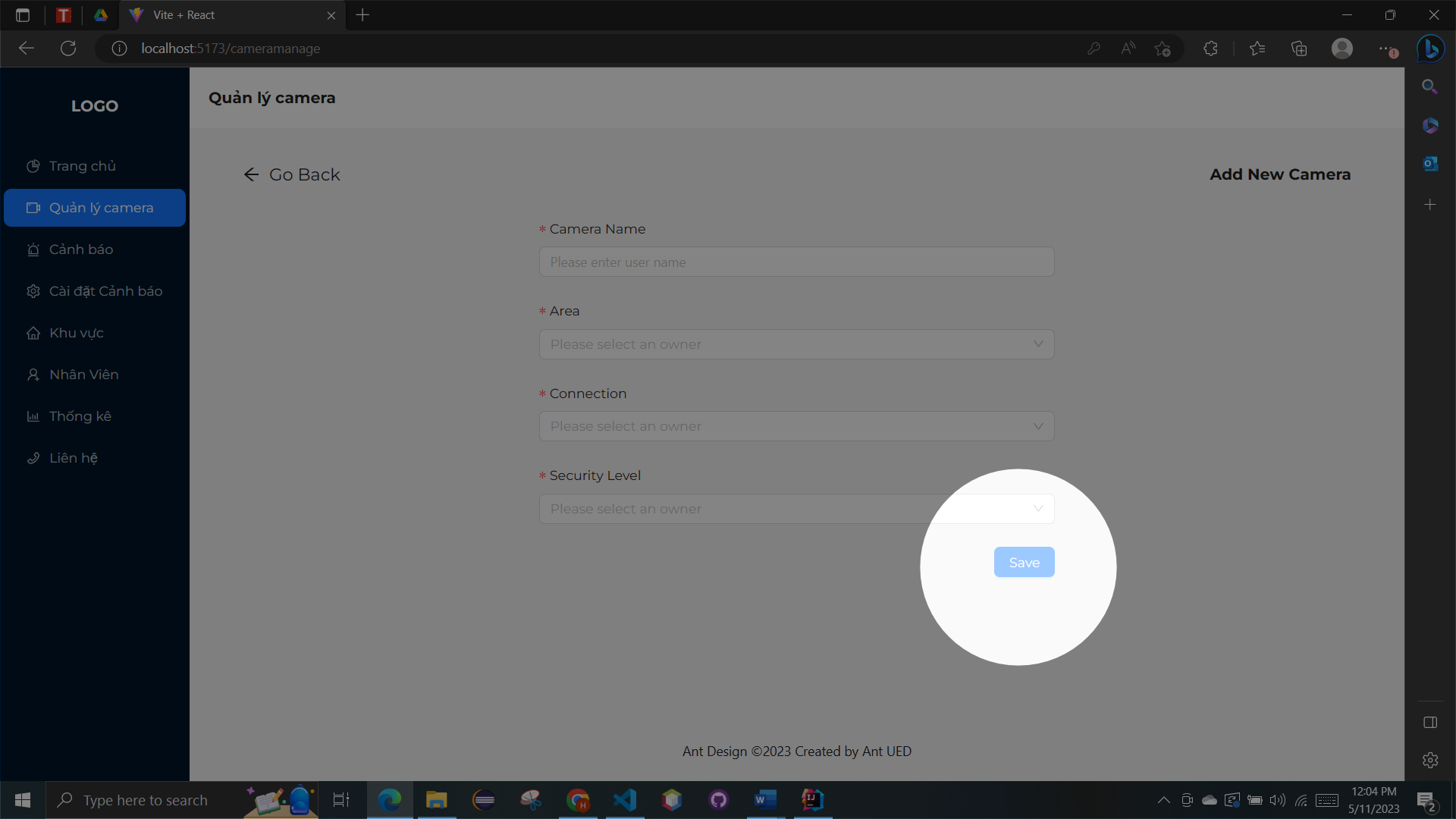
Figure 2.8.Trang thống kê lưu lượng cảnh báo

## Hướng dẫn sử dụng

- Ở trang login, nhập đúng tài khoản và mật khẩu đã được cung cấp để có thể vào hệ thống.

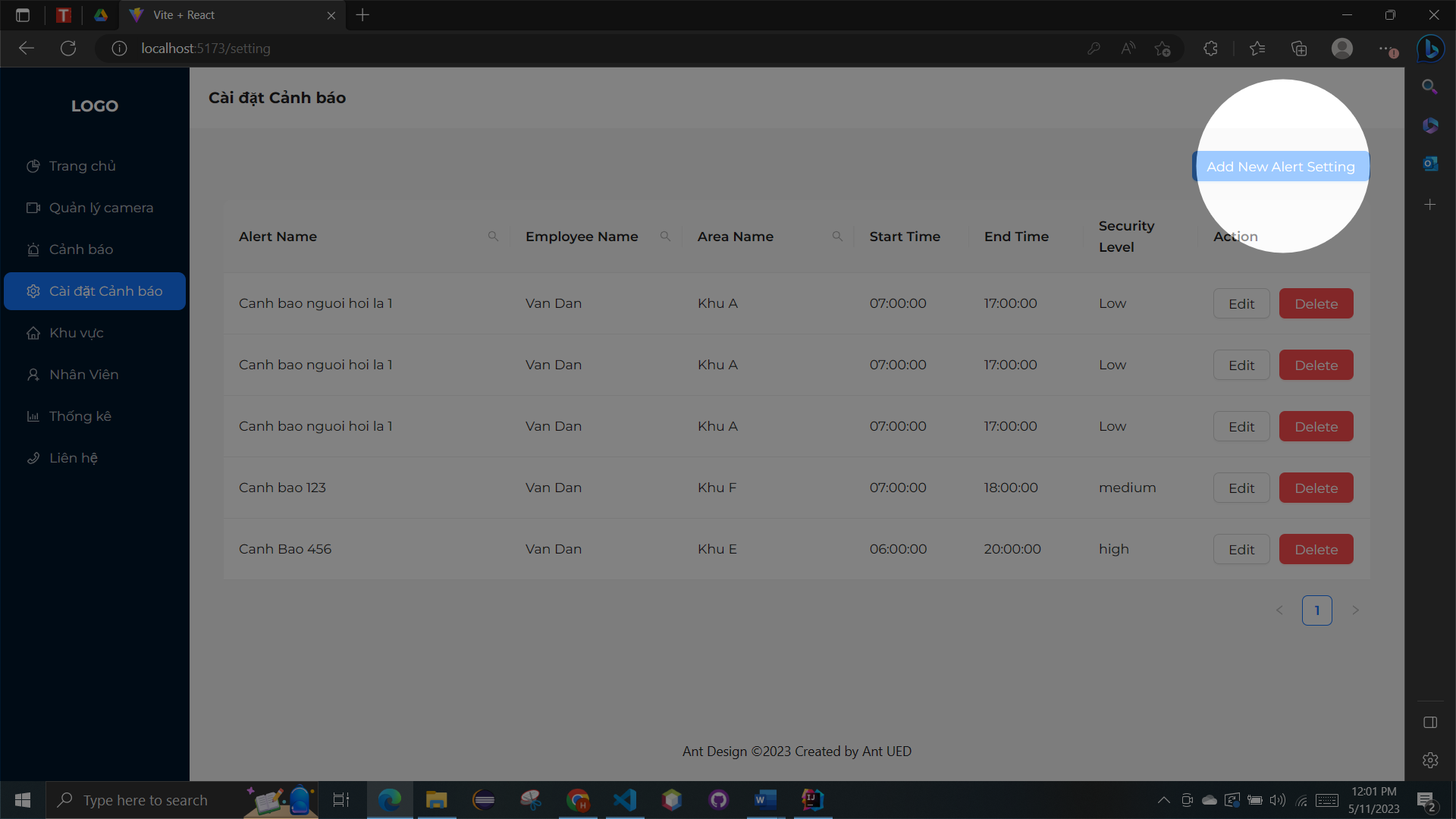
- Ở trang chủ, quản lý có thể xem các camera hiện đang hoạt động. Quản lý cũng có thể thêm một camera mới và gắn nó vào một khu vực chưa có camera:





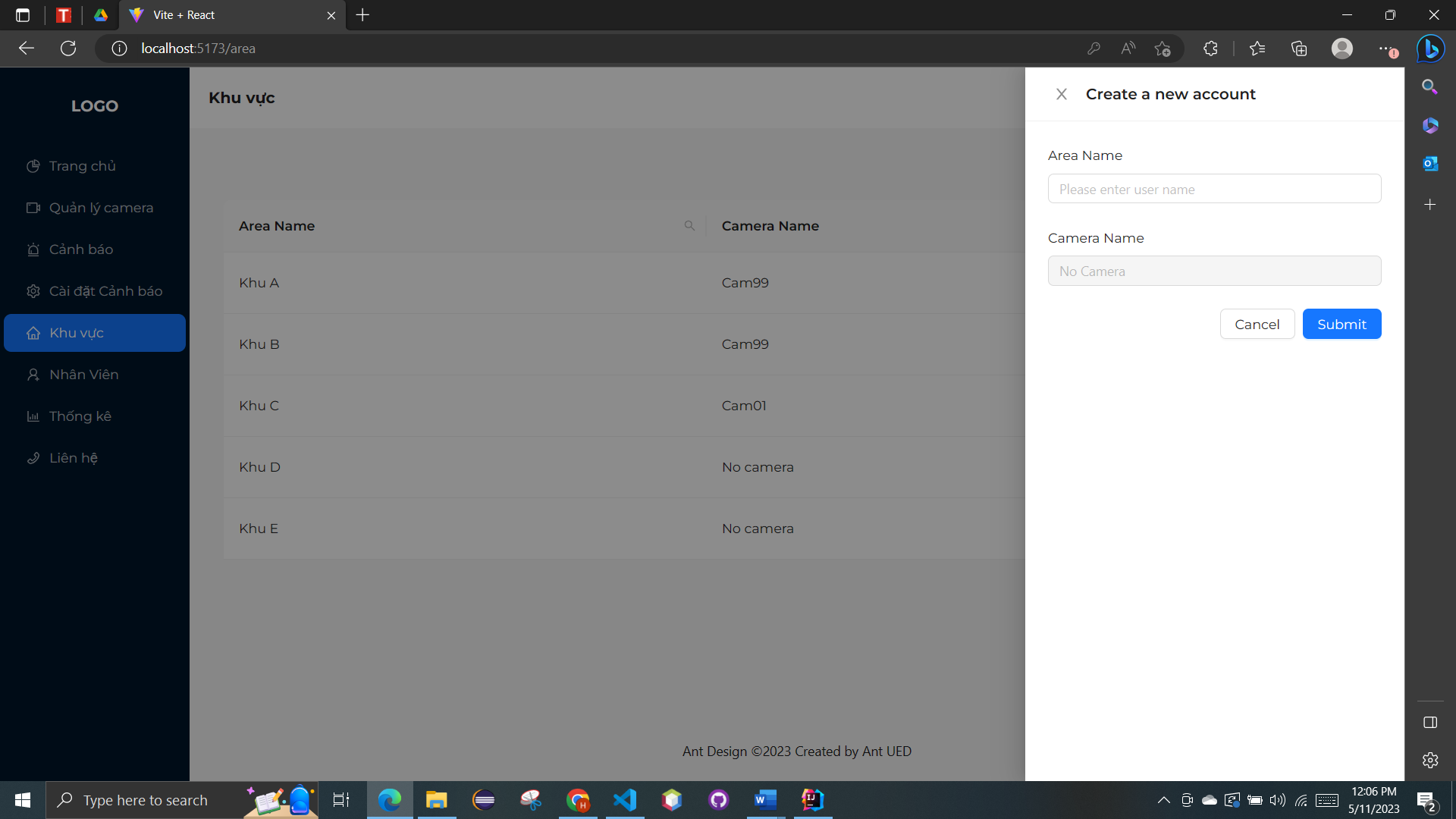
- Ở trang cảnh báo: đây là nơi các cảnh báo được gửi từ camera về hệ thống, cho biết thông tin về thời gian gửi cảnh báo, nhân viên cảnh báo, khu vực cảnh báo.

- Ở trang cài đặt cảnh báo: đây là trang lưu giữ các bộ cài đặt cảnh báo theo thời gian, nhân viên, khu vực, có nhiều bộ cài đặt được thiết lập bởi người dùng, khi camera phát hiện thông tin khớp với một trong các bộ cài đặt sẽ tiến hành gửi cảnh báo về hệ thống, có thể thêm bộ cài đặt mới bằng cách nhấn “Add new Alert Setting” và điền các thông tin cần thiết sau đó bấm “Save”:





- Ở trang khu vực: là nơi quản lý dùng để kiểm tra thông tin các khu vực hiện đang trong phận sự quản lý của công ty, có khu vực đã có Camera và một số thì chưa có, ở đây người quản lý có thể thêm khu vực mới đồng thời chỉnh sửa thông tin của khu vực.

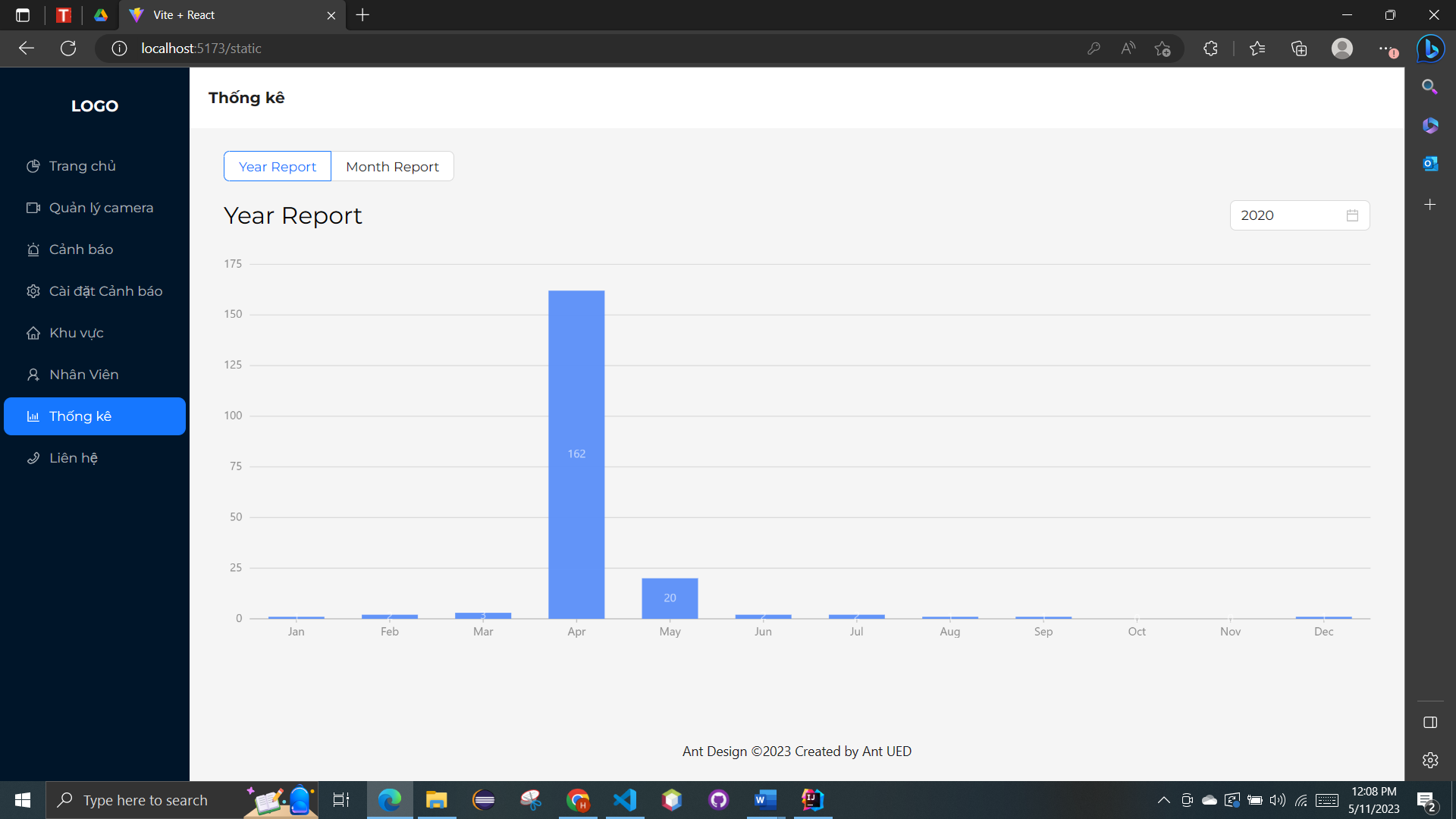


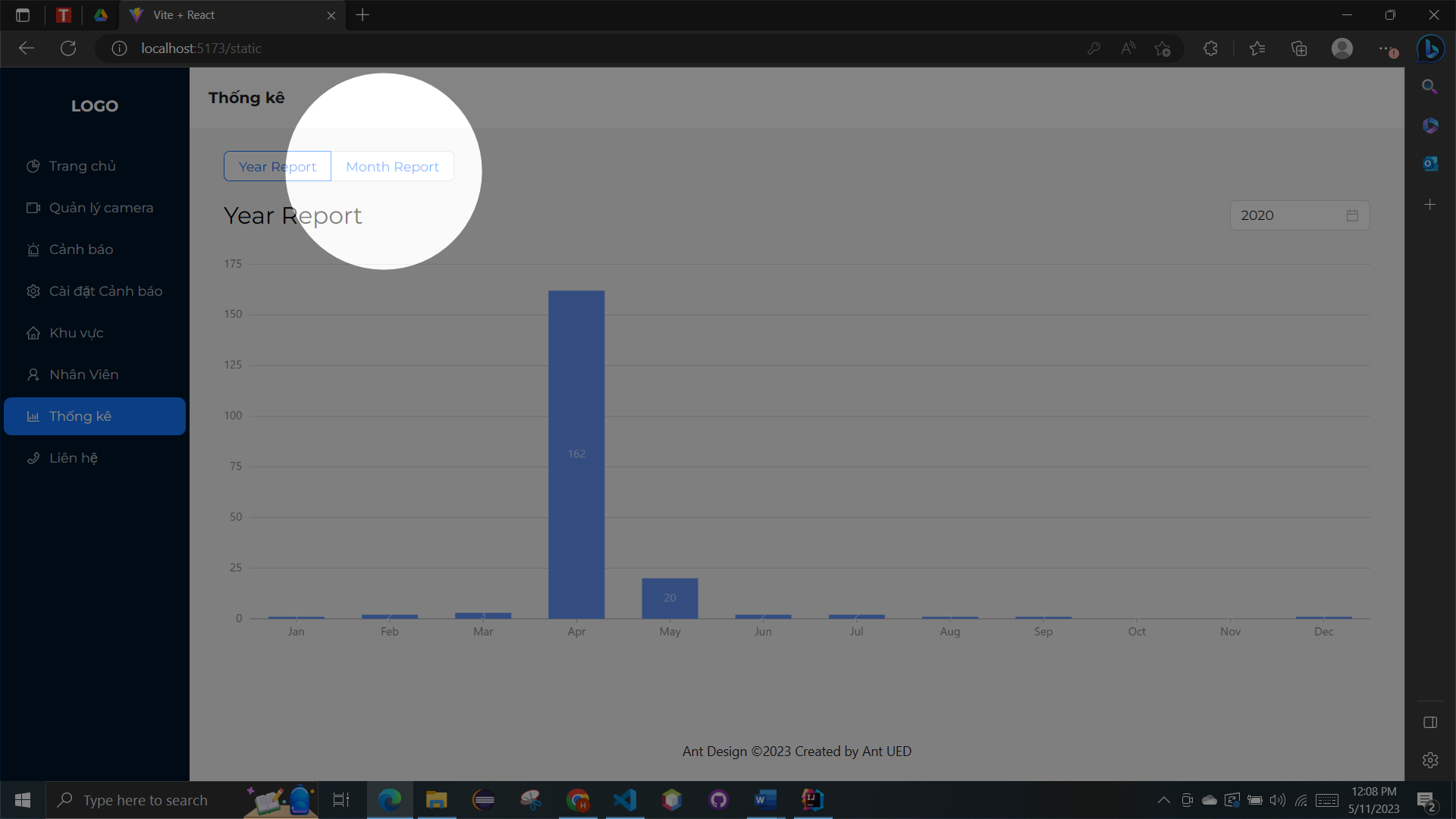
- Ở trang nhân viên: dùng để quản lý các nhân viên thuộc phận sự công ty, có thể thêm mới nhân viên, chỉnh sửa thông tin nhân viên:

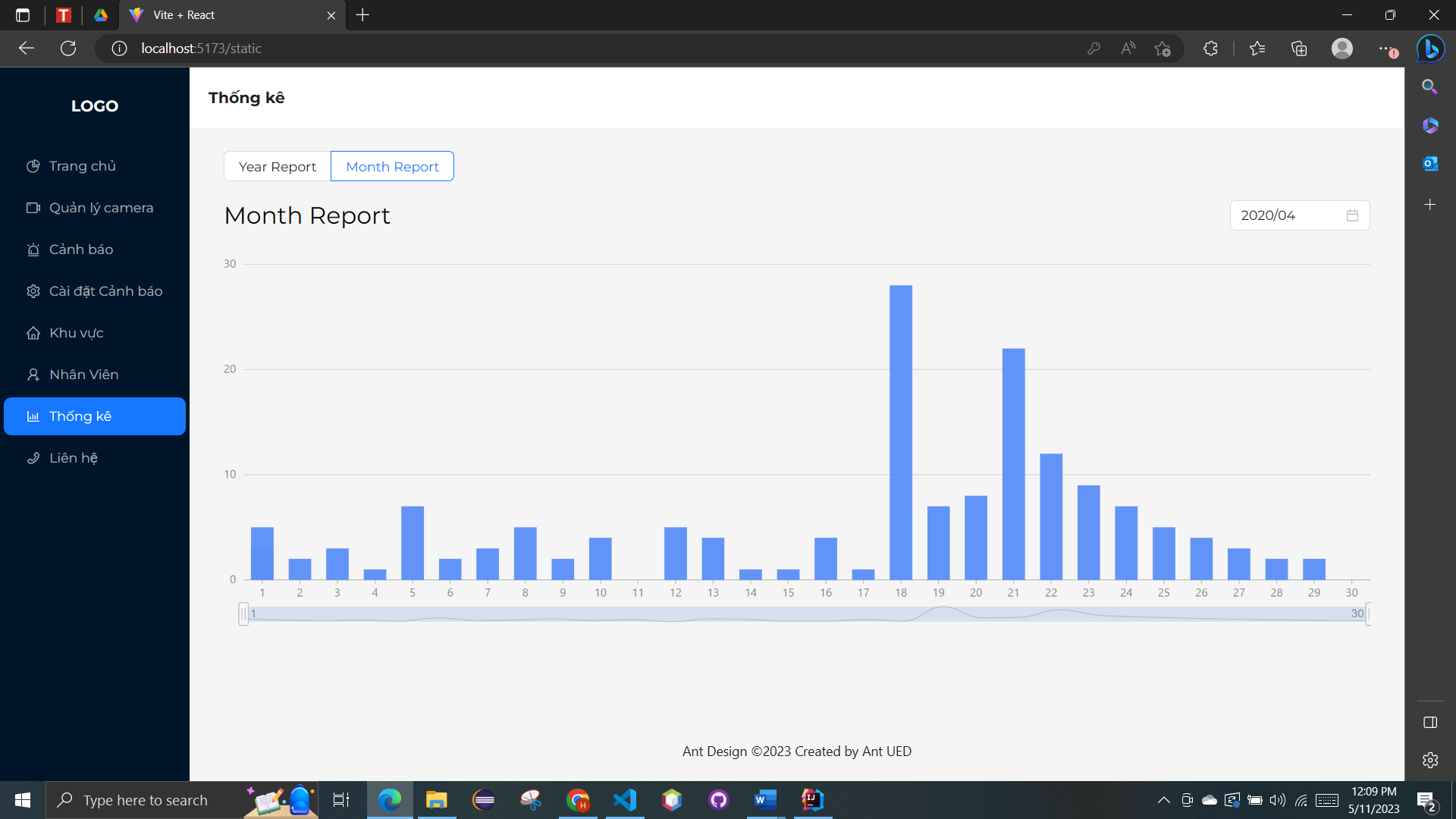




- Ở trang thống kê: dùng để kiểm tra, giám sát lưu lượng cảnh báo được gửi về hệ thống trong một khoảng thời gian, mà ở đây chính là theo các tháng trong một năm, hoặc có thể theo các ngày trong một tháng.







## Hướng dẫn deploy ứng dụng

Bước 1: Cài đặt git.

Bước 2: Cài đặt docker và docker compose.

Bước 3: Clone repository theo đường dẫn: <https://github.com/votranbaonguyen/AICameraDashboard.git>

Lệnh: **git clone** [**https://github.com/votranbaonguyen/AICameraDashboard.git**](https://github.com/votranbaonguyen/AICameraDashboard.git)

Bước 4: Di chuyển vào thư mục chứa dự án và file *docker-compose.yml*:

Lệnh: **cd AICameraDashboard/**

Bước 5: Chạy docker container:

Lệnh: ***docker-compose up*** hoặc ***docker-compose up -d***

Bước 6: Mở ứng dụng trên trình duyệt tại port 5173 để kiểm tra.

Mở trình duyệt tại với url: *http:****<ip>:****5173*

***<ip>*** là ip của máy đang chạy docker, nếu đang chạy trên local thì thay thế bằng ***localhost***

# Kết luận

## Kết quả đạt được

- Thấu hiểu tường tận cơ sở lý thuyết của việc phát triển một phần mềm theo hướng tiếp cận là hướng đối tượng, các quy trình làm việc thác nước, Agile.

- Áp dụng kiến thức trên vào việc xây dựng một hệ thống cụ thể, ở đây nhóm đã xây dựng thành công một hệ thống Dashboard giúp cho những người quản lý có thể quản lý các camera giám sát được gắn ở các khu vực của công ty khách hàng.

- Học được kỹ năng mềm cần thiết chính là làm việc nhóm: ở một dự án phần mềm thông thường, khối lượng công việc là tương đối lớn, chưa kể các kiến thức cần phải nắm cũng rất nhiều, thế nên cá nhân không thể đảm đương được mà phải cần đến một nhóm, để có thể làm việc chung thì kỹ năng giao tiếp, trình bày ý tưởng và quản lý dự án là vô cùng cần thiết.

## Định hướng phát triển

Ở dự án lần này, nhóm chỉ làm một hệ thống Dashboard dùng để quản lý camera, còn bản thân các camera có gắn thuật toán AI, học máy dùng để phát hiện, nhận diện và gửi cảnh báo ngược về hệ thống thì nhóm không có đảm nhiệm mà phụ thuộc vào một bên cung cấp API camera thứ 3. Vì tiêu chí của nhóm là tự thiết kế một phần mềm từ đầu đến cuối nên định hướng sẽ là có thể tự thiết kế một hệ thống nhận diện và phát triển tích hợp AI cho các camera.