|  |
| --- |
| **TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI**  **Viện Công Nghệ Thông Tin Và Truyền Thông**  **----- oOo -----** |
|  |
|  |
|  |
|  |
| **PROJECT 2** |
| **ĐỀ TÀI : Phần mềm Quản Lý Tài Sản cho doanh nghiệp** |
| Giảng viên hướng dẫn: ThS. Nguyễn Hồng Phương  Họ và tên sinh viên: Nguyễn Đức Mạnh  Mã số sinh viên: 201810445 |
| Lớp, khóa: VB2-CQ-CNTT- Khóa 2 |
| Hà Nội – Năm 2019 |

**TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI**

**Viện Công Nghệ Thông Tin Và Truyền Thông**

**----- oOo -----**

**PROJECT 2**

**ĐỀ TÀI: Phần mềm Quản Lý Tài Sản cho doanh nghiệp**

**Giảng viên hướng dẫn:** ThS. Nguyễn Hồng Phương

**Sinh viên:** Nguyễn Đức Mạnh

**MSSV:** 201810445

**Lớp:** KSVB2 – CQ – CNTT

**Năm học**: 2019-2020

Mục lục

[CHƯƠNG I: KHẢO SÁT DỰ ÁN 4](#_Toc26968915)

[***1.*** ***Động lực thực hiện dự án:*** 4](#_Toc26968916)

[***2.*** ***Phân tích khả thi dự án:*** 5](#_Toc26968917)

[CHƯƠNG II: PHÂN TÍCH HỆ THỐNG 6](#_Toc26968918)

[***1.*** ***Xác định yêu cầu*** 6](#_Toc26968919)

[***2.*** ***Phân tích ca sử dụng:*** 7](#_Toc26968920)

[***4.*** ***Biểu đồ luồng dữ liệu*** 19](#_Toc26968921)

[***4.*** ***Biểu đồ quan hệ*** 20](#_Toc26968922)

[CHƯƠNG III: THIẾT KẾ HỆ THỐNG 22](#_Toc26968923)

[***1.*** ***Thiết kế kiến trúc phần mềm:*** 22](#_Toc26968924)

[***2.*** ***Thiết kế giao diện:*** 23](#_Toc26968925)

[CHƯƠNG IV: CÀI ĐẶT PHẦN MỀM 31](#_Toc26968926)

[***1.*** ***Backend*** 31](#_Toc26968927)

[***2.*** ***Cài đặt thử nghiệm*** 33](#_Toc26968928)

[KẾT LUẬN 44](#_Toc26968929)

# CHƯƠNG I: KHẢO SÁT DỰ ÁN

1. ***Động lực thực hiện dự án:***

|  |
| --- |
| Yêu cầu hệ thống: Dự án quản lý vật tư, thiết bị online |
| Tài trợ dự án: Ngân sách công ty, nhà đầu tư |
| Nhu cầu công việc:   * Dự án được thực hiện để nâng cao hiệu quả công việc quản lý tài sản trong công ty. Khi hoạt động kinh doanh mở rộng, công ty cần quản lý số lượng tài sản lớn hơn. Nếu vẫn sử dụng cách thức ghi chép bằng sổ sách truyền thống sẽ kém hiệu quả và khả năng sai sót cao. |
| Yêu cầu thực tế: Hệ thống cần thực hiện những yêu cầu cơ bản sau   * Cơ sở dữ liệu về tài sản của công ty được lưu online và nhân viên quản lý tài sản có thể truy cập khi có kết nối internet * Lưu thông tin tài sản mới, truy vấn, sửa thông tin tài sản cũ, xóa tài sản không cần theo dõi * Thêm, sửa, xóa, truy vấn thông tin lịch sử sửa chữa, bảo trì của các tài sản * Phân quyền quản lý tài sản cho các nhân viên khác nhau * Cập nhật thông tin kiểm kê số lượng thực tế của các tài sản |
| Giá trị kinh tê:   * Nếu được đưa vào áp dụng, hệ thống ước tính sẽ giúp tiết kiệm 20% thời gian cho người quản lý tài sản |
| Yêu cầu khác:   * Thông tin về tài sản là thông tin bí mật của công ty do đó cần được lưu trữ bảo mật ở mức độ cao. * Thời gian thực hiện dự án là 3 tháng. |

1. ***Phân tích khả thi dự án:***

|  |  |  |
| --- | --- | --- |
| **Tiêu chí** | **Phân tích** | **Đánh giá độ rủi ro** |
| Mức độ quen thuộc của ứng dụng | Phầm mềm quản lý tài sản thuộc loại phần mềm quản lý, khá quen thuộc với người dùng. Có nhiều phần mềm tài chính, kế toán tương tự đã có trên thị trường như SAP, Misa | Thấp |
| Mức độ quen thuộc công nghệ | Phần mềm được xây dựng trên nền tảng web, dựa trên giao thức HTTP/TCP. Đây là 1 công nghệ đã tồn tại nhiều năm nên độ quen thuộc cao với lập trình viên. Tuy nhiên, dự án này dự định được xây dựng trên một framework mới như Django, dựa theo mô hình MVC, nên độ rủi ro sẽ hơi cao xét theo tiêu chí này. Tổng thể mức độ quen thuộc công nghệ ở mức trung bình | Trung bình |
| Quy mô dự án | Dự án được đánh giá ở mức nhỏ với khoảng 10 chức năng chính với khoảng hơn 1000 dòng lệnh | Thấp |
| Khá năng tương thích | Theo giả định, phần mềm mới được phát triển độc lập, không cần tích hợp vào các hệ thống cũ nên rủi ro này thấp | Thấp |

# CHƯƠNG II: PHÂN TÍCH HỆ THỐNG

1. ***Xác định yêu cầu***

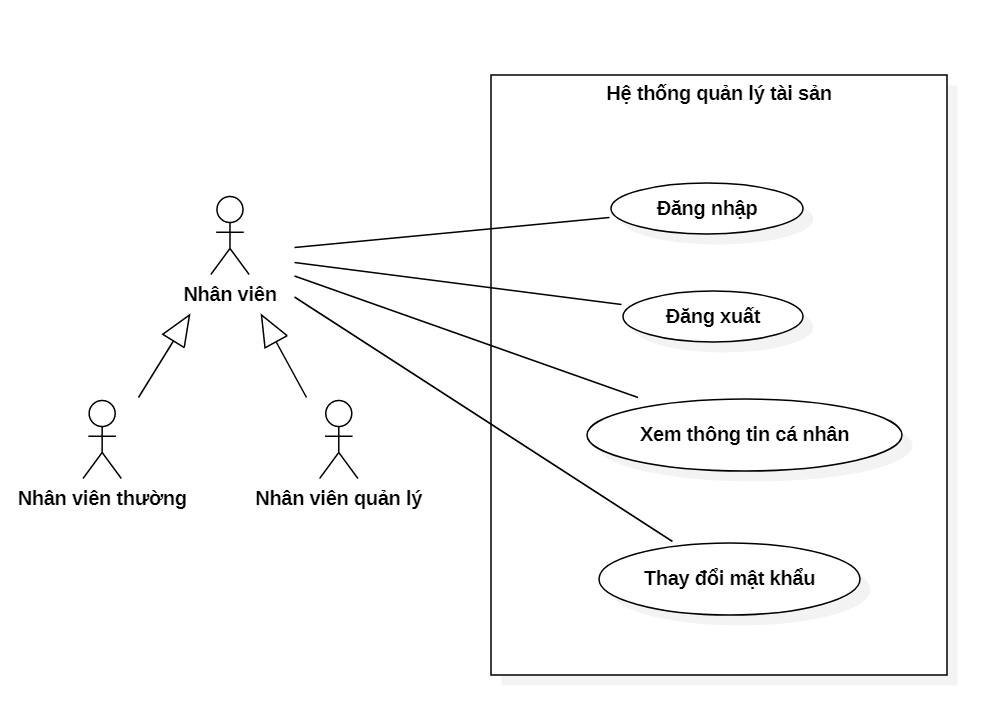
Phần mềm được thiết kế để quản lý các tài sản trong văn phòng. Các đối tượng cần lưu vào cơ sở dữ liệu là thông tin về các các tài sản bao gồm: mã tài sản, loại tài sản (vật tư hay thiết bị), tên tài sản, ngày bắt đầu sử dụng, số năm bảo hành, số năm dự định sử dụng, thiết bị cha chứa thiết bị cần lưu, hiện trạng sử dụng địa điểm đặt tài sản, người chịu trách nhiệm quản lý. Hệ thống cũng lưu thông tin về lịch sử bảo trì của tài sản: bao gồm ngày bảo trì, tài sản bảo trì, tiêu đề tóm tắt, và nội dung của công việc bảo trì. Cuối cùng, hệ thống cần lưu thông tin về các nhân viên tham gia quản lý tài sản.

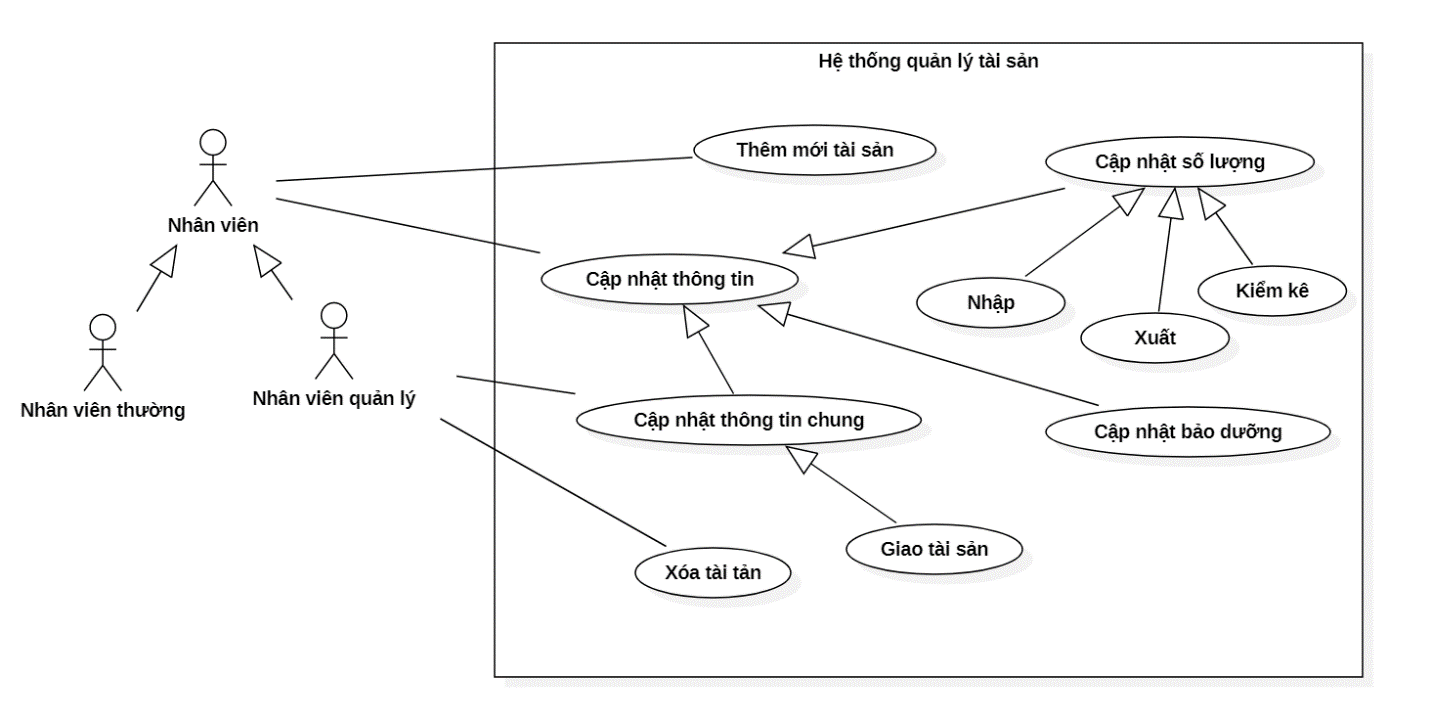
Các chức năng cần có:

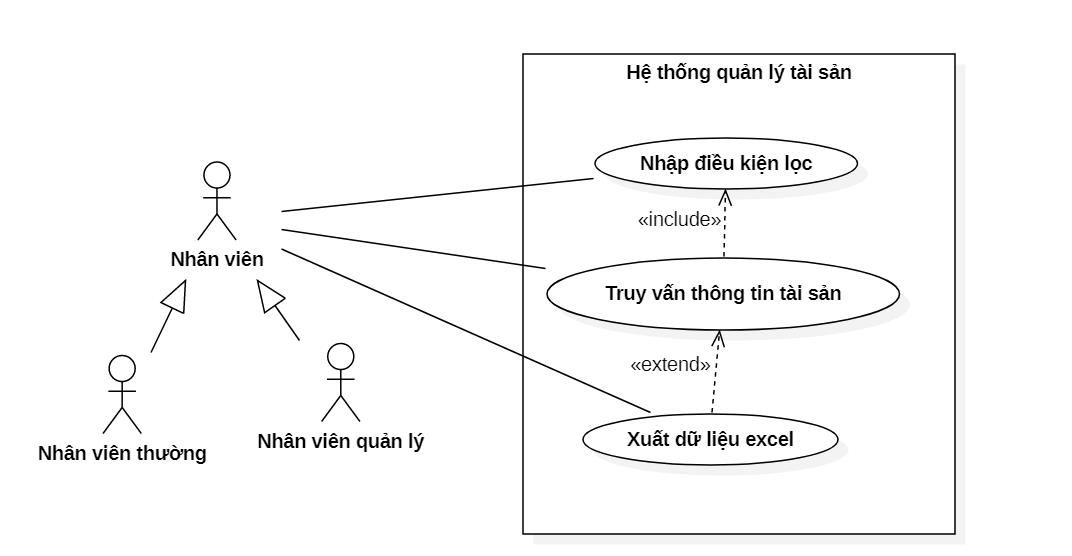
* 1. Tìm kiếm: Nhận vào dữ liệu từ người dùng về 1 trong các trường mã tài sản, loại tài sản hoặc tên tài sản. Phần mềm tiến hành lọc trong cơ sở dữ liệu để tìm ra các tài sản thỏa mãn điều kiện ban đầu và trả về danh sách cho người dùng.
  2. Xuất dữ liệu: cho phép người dùng sau khi tim kiếm tài sản có thể xuất kết quả và tải về dưới dạng file csv.
  3. Thêm tài sản mới: Người dùng nhập vào các thông tin về tài sản mới. Phần mềm sẽ tiến hành lưu vào cơ sở dữ liệu tài sản mới đó.
  4. Thêm thông tin bảo trì mới: Người dùng nhập vào các thông tin về bảo trì mới phát sinh. Phần mềm sẽ tiến hành lưu vào cơ sở dữ liệu bảo trì mới đó.
  5. Hiển thị thông tin chi tiết về tài sản: bao gồm thông tin về: mã tài sản, loại tài sản, tên tài sản, ngày bắt đầu sử dụng, số năm bảo hành, số năm dự định sử dụng và thiết bị cha, các thiết bị con, hiện trạng, địa điểm đặt tài sản, người quản lý và lịch sử bảo trì của thiết bị đó.
  6. Hiển thị ở trang chủ thông tin về các tài sản và bảo trì mới thêm gần đây.
  7. Cung cấp cho mỗi nhân viên quản lý tài sản 1 tài khoản để đăng nhập vào hệ thống.
  8. Người dùng có thể kiểm kê tình trạng hoạt động của tài sản và cập nhật vào cơ sở dữ liệu
  9. Mỗi nhân viên quản lý tài sản chỉ có thể nhìn thấy và cập nhật thông tin về tài sản của họ quản lý mà không thấy thông tin về các tài sản khác

1. ***Phân tích ca sử dụng:***

Dựa vào những yêu cầu ban đầu, ta vẽ được biểu đồ ca sử dụng của phần mềm như sau:







Biểu đồ 1.1: Biểu đồ ca sử dụng của phần mềm

*2.1 Mô tả ca sử dụng:*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Ca sử dụng | Đăng nhập | | | Mã ca : UC1 | |
| Tác nhân | Nhân viên quản lý tài sản | | | | |
| Mô tả | Nhân viên cần đăng nhập vào hệ thống để thực hiện các tác vụ cần thiết | | | | |
| Điều kiện trước | Không | | | | |
| Trường hợp thông thường | 1. Hệ thống hiển thị màn hình đăng nhập, yêu cầu người dùng nhập thông tin về tài khoản, mật khẩu 2. Người dùng nhập thông tin tài khoản 3. Hệ thống trả về màn hình chính, hiện thị các chức năng tiếp theo cho người dùng lựa chọn | | | | |
| Trường hợp ngoại lệ | 3a. Thông tin người dùng nhập sai, hệ thống trả về màn hình đăng nhập cùng dòng thông báo lỗi. Yêu cầu người dùng nhập lại | | | | |
| Tóm tắt | | | | | |
| Dữ liệu vào:  Tài khoản, mật khẩu | | Nguồn:  Người dùng | Dữ liệu ra:  -Màn hình chính  -Màn hình đăng nhập, thông báo lỗi | | Đích:  Người dùng |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Ca sử dụng | Thêm tài sản | | | Mã ca: UC2 | |
| Tác nhân | Nhân viên quản lý tài sản | | | | |
| Mô tả | Nhân viên thêm 1 tài sản mới vào cơ sở dữ liệu bằng cách nhập thông tin về tài sản mới vào hệ thống | | | | |
| Điều kiện trước | Nhân viên được xác thực | | | | |
| Trường hợp thông thường | 1. Hệ thống hiển thị màn hình với các trường thông tin về tài sản trống, yêu cầu người dùng nhập các thông tin này 2. Người dùng nhập thông tin tài sản 3. Hệ thống lưu thông tin tài sản vào cơ sở dữ liệu và trả về màn hình thông báo thêm tài sản thành công, cho phép người dùng chọn nhập tiếp hoặc về màn hình chính | | | | |
| Trường hợp ngoại lệ | 3a. Khi người dùng nhập thiếu thông tin bắt buộc, hệ thống yêu cầu người dùng điền đủ thông tin đó | | | | |
| Tóm tắt | | | | | |
| Dữ liệu vào:  Các trường thông tin tài sản | | Nguồn:  Người dùng | Dữ liệu ra:  -Thông báo thành công  -Dữ liệu tài sản  - thông báo lỗi | | Đích:  -Người dùng  -Cơ sở dữ liệu |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Ca sử dụng | Thêm thông tin bảo trì | | | Mã ca: UC 3 | |
| Tác nhân | Nhân viên quản lý tài sản | | | | |
| Mô tả | Nhân viên thêm thông tin về 1 lần bảo trì mới vào cơ sở dữ liệu | | | | |
| Điều kiện trước | Nhân viên được xác thực | | | | |
| Trường hợp thông thường | 1. Hệ thống hiển thị màn hình với các trường thông tin về bảo trì trống, yêu cầu người dùng nhập các thông tin này 2. Người dùng nhập thông tin bảo trì 3. Hệ thống lưu thông tin bảo trì vào cơ sở dữ liệu và trả về màn hình thông báo thêm tài sản thành công, cho phép người dùng chọn nhập tiếp hoặc về màn hình chính | | | | |
| Trường hợp ngoại lệ | 3a. Khi người dùng nhập thiếu thông tin bắt buộc, hệ thống yêu cầu người dùng điền đủ thông tin đó | | | | |
| Tóm tắt | | | | | |
| Dữ liệu vào:  Các trường thông tin bảo trì | | Nguồn:  Người dùng | Dữ liệu ra:  -Thông báo thành công  -Dữ liệu bảo trì  - thông báo lỗi | | Đích:  -Người dùng  -Cơ sở dữ liệu |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Ca sử dụng | Tìm kiếm tài sản | | | Mã ca : UC4 | |
| Tác nhân | Nhân viên quản lý tài sản | | | | |
| Mô tả | Nhân viên tìm 1 hoặc 1 nhóm tài sản thỏa mãn điều kiện cho trước bằng cách nhập thông tin về tài sản vào hệ thống | | | | |
| Điều kiện trước | Nhân viên được xác thực | | | | |
| Trường hợp thông thường | 1. Hệ thống hiển thị màn hình với các trường thông tin về điều kiện tài sản, yêu cầu người dùng nhập các thông tin này 2. Người dùng nhập các điều kiện về mã tài sản, loại tài sản, tên tài sản, ngày bắt đầu sử dụng trong khoảng thời gian, hiện trạng tài sản 3. Hệ thống tìm kiếm các tài sản thỏa mãn tất các các điều kiện lọc trong cơ sở dữ liệu và trả về màn liệt kê các tài sản thỏa mãn, cho phép người dùng chọn vào 1 tài sản để xem chi tiết. | | | | |
| Trường hợp thay thế | 1. Nêu không tìm thấy kết quả phù hợp, hệ thống trả về thông báo cho người dùng | | | | |
| Tóm tắt | | | | | |
| Dữ liệu vào:  Các điều kiện lọc | | Nguồn:  Người dùng | Dữ liệu ra:  -Danh sách tài sản  - thông báo lỗi | | Đích:  -Người dùng |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Ca sử dụng | Kiểm kê tài sản | | | Mã ca: UC5 | |
| Tác nhân | Nhân viên quản lý tài sản | | | | |
| Mô tả | Nhân viên cập nhật hiện trạng của tài sản bằng cách cách nhập mã tài sản và hiện trạng tài sản vào hệ thống | | | | |
| Điều kiện trước | Nhân viên được xác thực | | | | |
| Trường hợp thông thường | 1. Hệ thống hiện 1 form yêu cầu người dùng nhập mã tài sản và trạng thái cần kiểm kê 2. Người dùng nhập mã tài sản cần kiểm kê vào hệ thống 3. Hệ thống trả về tên tài sản, loại tài sản, ngày bắt đầu sử dụng 4. Người dùng kiểm tra thông tin tài sản hệ thống trả về tài sản hệ thống 5. Nếu thông tin đúng, người dùng tiến hành cập nhật trạng thái tài sản, nếu thông tin sai người dùng kiểm tra lại mã tài sản và nhập lại từ bước 2 6. Sau khi nhập xong tất cả số tài sản muốn kiểm kê, người dùng bấm nút cập nhật để cập nhật toàn bộ thông tin kiểm kê vào cơ sở dữ liệu 7. Hệ thống trả về thông báo cập nhật thành công cho người dùng | | | | |
| Trường hợp thay thế | 1. Nếu form hết số dòng để cập nhật, người dùng nhập số dòng muốn thêm 2. Hệ thống thêm số dòng người dùng nhập mới vào form 3. Người dùng tiếp tục nhập thông tin tài sản kiểm kê mới từ bước 2 | | | | |
| Trường hợp ngoại lệ | 7. Nếu tài sản cập nhật không có trong hệ thống do người dùng nhập sai mã, hệ thông sẽ gửi lại báo lỗi cập nhật tương ứng với tài sản đó. | | | | |
| Tóm tắt | | | | | |
| Dữ liệu vào:  -Các mã tài sản  -Các trạng thái tương ứng với tài sải | | Nguồn:  -Người dùng | Dữ liệu ra:  -Thông báo thành công  - thông báo lỗi  -Dữ liệu cập nhật | | Đích:  -Người dùng  -Cơ sở dữ liệu |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Ca sử dụng | Xuất dữ liệu csv | | | Mã ca : UC6 | |
| Tác nhân | Nhân viên quản lý tài sản | | | | |
| Mô tả | Nhân viên xuất dữ liệu về tài sản dưới dạng csv | | | | |
| Điều kiện trước | Nhân viên được xác thực | | | | |
| Trường hợp thông thường | 1. Nhân viên nhập các điều kiện về tài sản cần kết xuất giốn như UC3  2. Hệ thống trả về danh sách cách tài sản thỏa mãn điều kiện  3. Nhân viên chọn nút kết xuất  4. Hệ thống trả về file csv cho người dùng lưu vào máy của họ | | | | |
| Tóm tắt | | | | | |
| Dữ liệu vào:  Các điều kiện lọc | | Nguồn:  Người dùng | Dữ liệu ra:  -Danh sách tài sản  - File csv thông tin các tài sản thỏa mãn | | Đích:  -Người dùng |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Ca sử dụng | Xem chi tiết tài sản | | | Mã ca : UC7 | |
| Tác nhân | Nhân viên quản lý tài sản | | | | |
| Mô tả | Nhân viên muốn xem thông tin chi tiết về 1 tài sản trong danh sách họ quản lý | | | | |
| Điều kiện trước | Nhân viên được xác thực | | | | |
| Trường hợp thông thường | 1. Nhân viên nhập các điều kiện về tài sản cần kết xuất giống như UC3  2. Hệ thống trả về danh sách cách tài sản thỏa mãn điều kiện  3. Nhân viên chọn đường link tương ứng với tài sản họ muốn xem chi tiết  4. Hệ thống hiển thị thông tin chi tiết về tài sản đó | | | | |
| Tóm tắt | | | | | |
| Dữ liệu vào:  Các điều kiện lọc | | Nguồn:  Người dùng | Dữ liệu ra:  -Danh sách tài sản  - Thông tin chi tiết về 1 tài sản | | Đích:  -Người dùng |

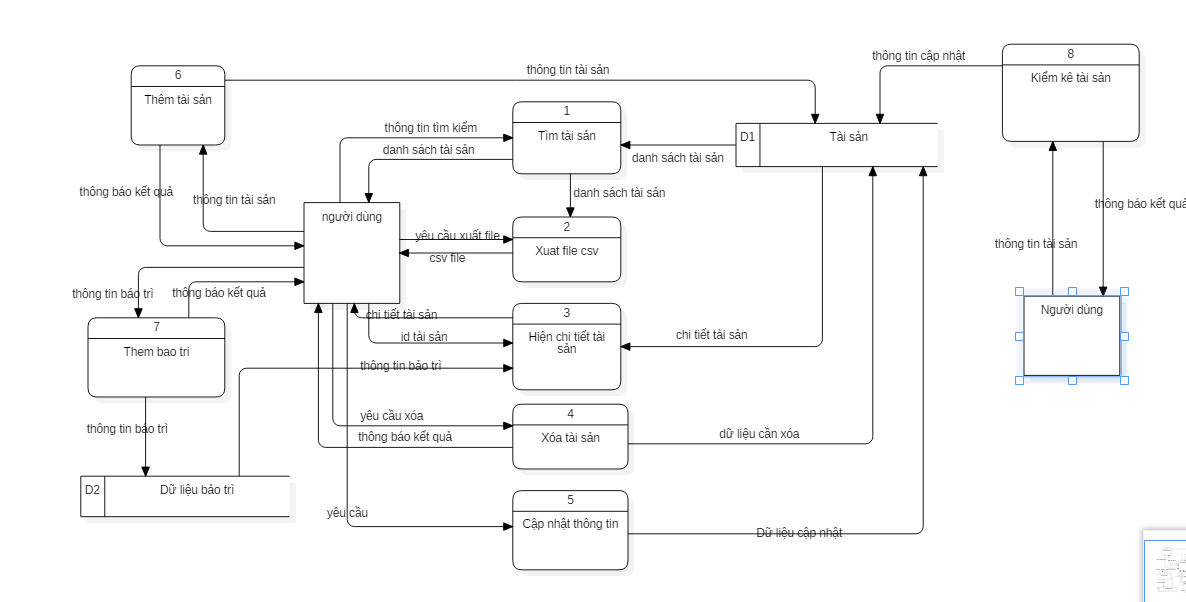
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Ca sử dụng | Xóa tài sản | | | Mã ca : UC8 | |
| Tác nhân | Trưởng phòng quản lý tài sản (Asset Manager) | | | | |
| Mô tả | Nhân viên muốn xem thông tin chi tiết về 1 tài sản trong danh sách họ quản lý | | | | |
| Điều kiện trước | Nhân viên được xác thực | | | | |
| Trường hợp thông thường | 1. Người dùng thực hiện chức năng xem chi tiết như UC6 2. Người dùng chọn chức năng xóa tài sản ở màn hình thông tin chi tiết 3. Hệ thống trả về cảnh báo hỏi người dùng có chắc chắn muốn xóa tài sản 4. Người dùng xác nhận yêu cầu xóa 5. Hệ thống xóa tài sản và trả về thông báo xóa thành công | | | | |
| Tóm tắt | | | | | |
| Dữ liệu vào:  Các điều kiện lọc | | Nguồn:  Người dùng | Dữ liệu ra:  -Danh sách tài sản  - Thông tin chi tiết về 1 tài sản | | Đích:  -Người dùng |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Ca sử dụng | Cập nhật thông tin tài sản | | | Mã ca : UC9 | |
| Tác nhân | Nhân viên tổng quản lý tài sản (Asset Manager) | | | | |
| Mô tả | Nhân viên quản lý muốn cập thật thông tin về 1 tài sản | | | | |
| Điều kiện trước | Nhân viên được xác thực | | | | |
| Trường hợp thông thường | 1. 1 Người dùng thực hiện chức năng xem chi tiết như UC6 2. Người dùng chọn chức năng cập nhật thông tin tài sản ở màn hình thông tin chi tiết 3. Hệ thống trả về form thông tin tài sản với các trường đã được điền sẵn bởi các giá trị hiện tại của tài sản 4. Người dùng thay đổi nội dung các trường trong form sau đó chọn cập nhật 5. Hệ thống trả về thông báo cập nhật thành công | | | | |
| Tóm tắt | | | | | |
| Dữ liệu vào:  Các điều kiện lọc  Các thông tin cập nhật | | Nguồn:  Người dùng | Dữ liệu ra:  -Danh sách tài sản  - Form cập nhật thông tin | | Đích:  -Người dùng |

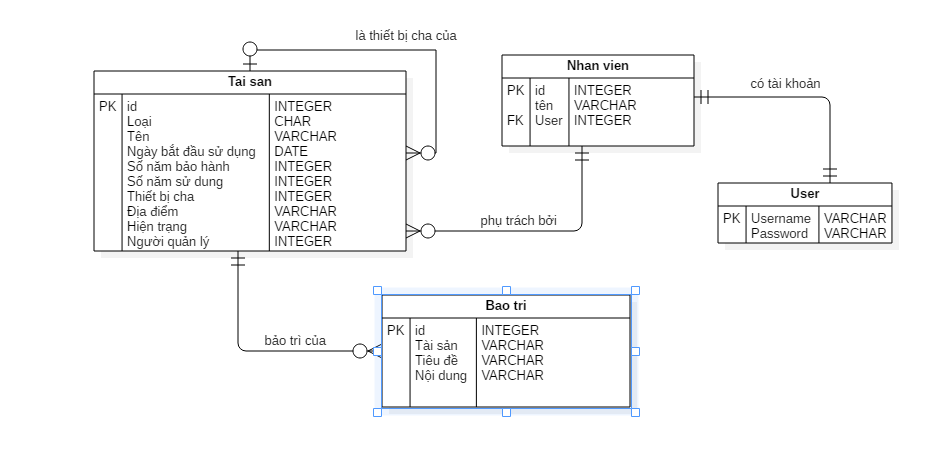
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Ca sử dụng | Cập nhật thông tin tài sản | | | Mã ca : UC10 | |
| Tác nhân | Nhân viên quản lý tài sản | | | | |
| Mô tả | Nhân viên quản lý muốn cập thật thông tin về 1 tài sản | | | | |
| Điều kiện trước | Nhân viên được xác thực  Tài sản được giao cho nhân viên quản lý | | | | |
| Trường hợp thông thường | 1. Nhân viên yêu cầu thêm dòng mới vào form 2. Hệ thống hiển thị giao diện tìm kiếm tài sản muốn thêm 3. Nhân viên nhập chuỗi kí tự tên tài sản cần tìm kiếm 4. Hệ thống trả về danh sách các tài sản nhân viên có thể cần 5. Nhân viên chọn 1 tài sản cần điền vào form cập nhật 6. Nhân viên nhập số lượng cần nhập xuất 7. Nhân viên gửi yêu cầu cập nhật giao dịch tới hệ thống 8. Hệ thống thông báo giao dịch được ghi thành công vào cơ sở dữ liệu | | | | |
| Tóm tắt | | | | | |
| Dữ liệu vào:  Các điều kiện lọc  Các thông tin cập nhật | | Nguồn:  Người dùng | Dữ liệu ra:  -Danh sách tài sản  - Form cập nhật thông tin | | Đích:  -Người dùng |

1. ***Biểu đồ luồng dữ liệu***

Từ những phân tích đặc tả ca sử dụng, ta vẽ được sơ đồ luồng dữ liệu mức 0 như sau:



1. ***Biểu đồ quan hệ***



***Giải thích cơ sở dữ liệu:***

* Tài Sản: Theo yêu cầu lưu trữ thông tin, mỗi tài sản có thể là 1 thiết bị đơn lẻ ví dụ như bút, giấy, đồng hồ… nhưng cũng có thể là một tập hợp gồm nhiều tài sản khác ví dụ như 1 chiếc máy tính bao gồm chuột, bàn phím, ram, cpu…Do đó mỗi tài sản tồn tại 1 liên kết là thiết bị con với 0 hoặc 1 tài sản khác. Đồng thời nó cũng là cha của 0 hoặc nhiều tài sản khác.
* Bảo trì: mỗi bản ghi bảo trì là nội dung của 1 lần bảo trì của 1 tài sản nào đó. Do đó giữa dữ liệu bảo trì và Tài sản có mối liên kết n-1: mối tài sản có nhiều lần bảo trì, mỗi bản ghi bảo trì là của 1 tài sản.
* Nhân viên:
* Mỗi tài sản được quản lý bởi 1 nhân viên trong hệ thống. Do đó nhân viên và tài sản sẽ tồn tại liên kết 1-n: mỗi tài sản phụ trách bởi 1 nhân viên, ngược lại mỗi nhân viên phụ trách nhiều tài sản.
* Django Frame work cung cấp 1 cơ sở dữ liệu user phục vụ cho việc xác thực thông tin người dùng. Để thiết kế mỗi nhân viên được xác thực, ta cần 1 liên kết 1-1 giữa nhân viên và user.

Từ những phân tích trên ta thiết kế được các bảng trong cơ sở dữ liệu như sau:

Tài sản (Mã tài sản, Loại, Tên, Ngày bắt đầu sử dụng, Số năm bảo hành, số năm sử dụng, *Thiết bị cha,* Địa điểm, hiện trạng, *Mã nhân viên*)

Bảo trì (Mã bảo trì, *Mã tài sản*, tiêu đề, nội dung)

Nhân viên (Mã Nhân viên, tên nhân viên, *user ID*)

User (Mã user, username, password)

# CHƯƠNG III: THIẾT KẾ HỆ THỐNG

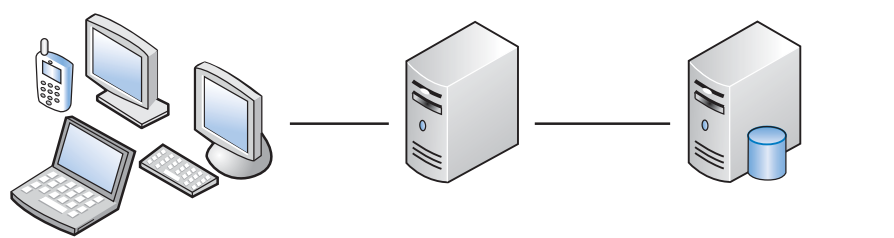
1. ***Thiết kế kiến trúc phần mềm:***

Kiến trúc phần mềm được thiết kế dựa trên phân tích các yêu cầu phi chức năng của hệ thống như sau:

|  |  |
| --- | --- |
| **Yêu cầu hoạt động** | |
| Yêu cầu môi trường kĩ thuật | * Hệ thống sẽ phải hoạt động trên môi trường web với các trình duyệt thông dụng như google chrome, mozila firefox, IE… * Người dùng chỉ cần có trình duyệt web trên máy tính của mình để sử dụng hệ thống |
| Yêu cầu tích hợp hệ thống | * Hệ thống cần đọc và ghi vào cơ sở dữ liệu tài sản của doanh nghiệp bằng cách tương tác qua 1 hệ quản trị cơ sở dữ liệu trung gian |
| Yêu cầu tính di động | * Hệ thống cần làm việc được trên các trình duyệt web khác nhau như chrome, firefox, IE… và trên các hệ điều Linux và Windows |
| Yêu cầu tính bảo trì | * Hệ thống được thiết kế là 1 ứng dụng được tích hợp trong tương lai vào bộ các ứng dụng quản trị doanh nghiệp bao gồm các ứng dụng như quản lý nhân sự, tài chính, bán hàng… |
| **Yêu cầu hiệu suất** | |
| Yêu cầu tốc độ | * Thời gian phản hồi của hệ thống nhỏ hơn 9s |
| Yêu cầu sức chứa | * Hệ thống cần được thiết kế để sử dụng bởi 5-10 nhân viên quản lý tài sản cùng lúc |
| Yêu cầu tính sẵn sàng và tin cậy | * Hệ thống cần phải sẵn sàng hoạt động 24/7 |
| **Yêu cầu bảo mật** | |
| Giá trị ước tính của hệ thống | * Tổng giá trị các tài sản được lưu trong hệ thống ước tính 500 triệu tới 1 tỉ đồng * Thiệt hại nếu hệ thống bị hỏng hoặc xảy ra mất mát dữ liệu sẽ là rất lớn với doanh nghiệp |
| Yêu cầu kiểm soát truy cập hệ thống | * Nhân viên sử dụng tên đăng nhập và password để đăng nhập vào hệ thống * Chỉ trưởng phòng quản lý tài sản (asset manager) mới có thể thay đổi thông tin tài sản và giao cho nhân viên khác quản lý * Các nhân viên chỉ có thể cập nhật số lượng, hiện trạng theo dõi của tài sản |
| Yêu cầu mã hóa , xác thực | * Các thông tin về tài sản trong doanh nghiệp và mật khẩu tài khoản đăng nhập của người dùng cần được mã hóa và lưu trữ bảo mật * Hệ thống cần có cơ chế bảo vệ lại các cuộc tấn công, đánh cắp dữ liệu trên mạng |

Từ những yêu cầu phi chức năng trên, ta thấy kiến trúc hợp lý nhất cho hệ thống là kiến trúc thin client-server với 4 thành phần chính như sau:

* *Cơ sở dữ liệu:* Nơi lưu trữ các dữ liệu được thiết kế theo các mô hình mà hệ thống phần mềm cần sử dụng
* *Thành phần truy xuất dữ liệu:* là 1 hệ quản trị cơ sở dữ liệu(DBMS) được dùng để đọc và ghi dữ liệu tới cơ sở dữ liệu
* *Thành phần ứng dụng:* là thành phần trung gian, nhận các yêu cầu, dữ liệu từ người dùng sau đó xử lý, tính toán, thực hiện các thao tác đọc ghi, với thành phần truy xuất dữ liệu, và cuối cùng trả về kết quả cho người dùng.
* *Thành phần hiển thị:* Có nhiệm vụ hiển thị giao diện ứng dụng, và trình bày các các thông tin người cần

**

Máy chủ dữ liệu

Cơ sở dữ liệu

Hệ quản trị CSDL

Máy khách(client)

Thành phần hiển thị

Máy chủ ứng dụng

Thành phần ứng dụng

Trong kiến trúc thin client-server, thành phần hiển thị được chạy trên thiết bị người dùng (trình duyệt web), thành phần ứng dụng được chạy trên máy chủ, và thành phần cơ sở dữ liệu và thành phần truy xuất dữ liệu được chạy trên 1 máy chủ khác hoặc có thể cùng máy chủ với thành phần ứng dụng.

1. ***Thiết kế giao diện:***

2.1 Phân tích tình huống sử dụng

2.2 Thiết kế cấu trúc giao diện

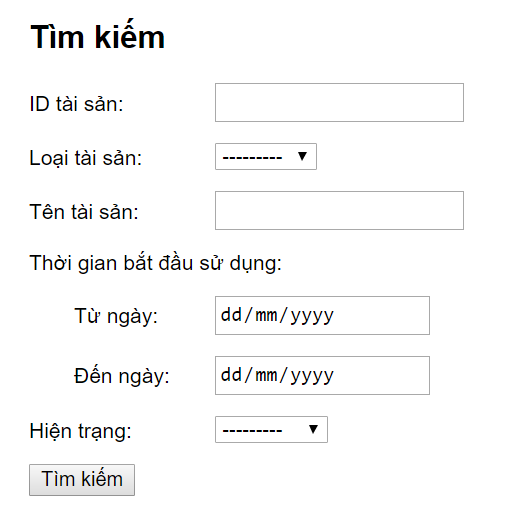
* 1. Thiết kế nguyên mẫu giao diện:

*Giao diện trang chủ:*



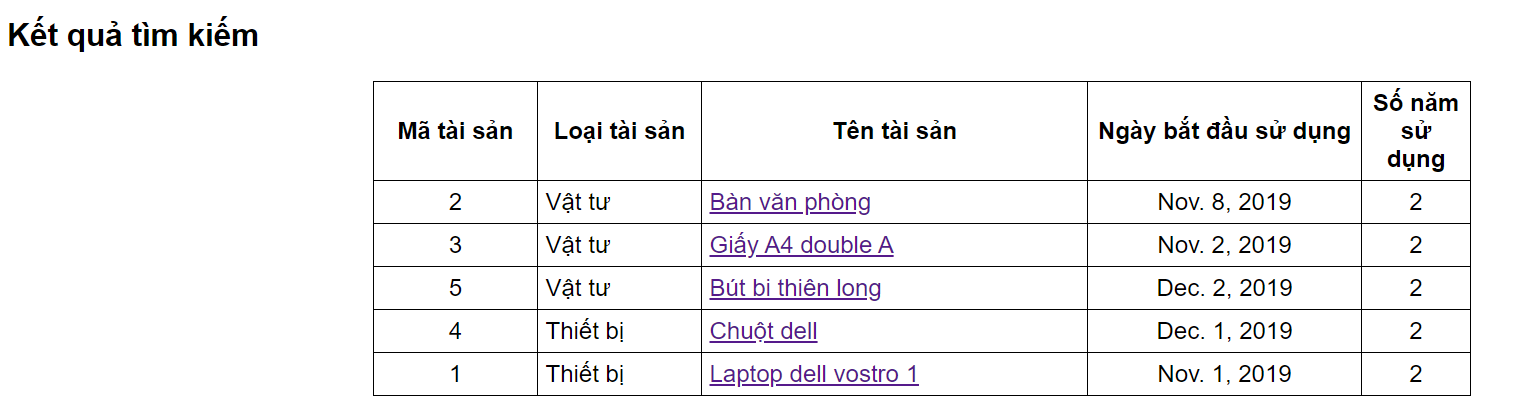
Giao diện trang chủ chứa đường dẫn tới các chức năng chính, thường được dùng của hệ thống liên quan đến cơ sở dữ liệu tài sản như tìm kiếm, truy vấn thông tin, cập nhật số lượng nhập, xuất, kiểm kê. Ngoài ra nó còn có các đường dẫn tới các chức năng xem thông tin cá nhân, đổi mật khẩu cho tài khoản và đăng xuất.

*Giao diện tìm kiếm tài sản:*



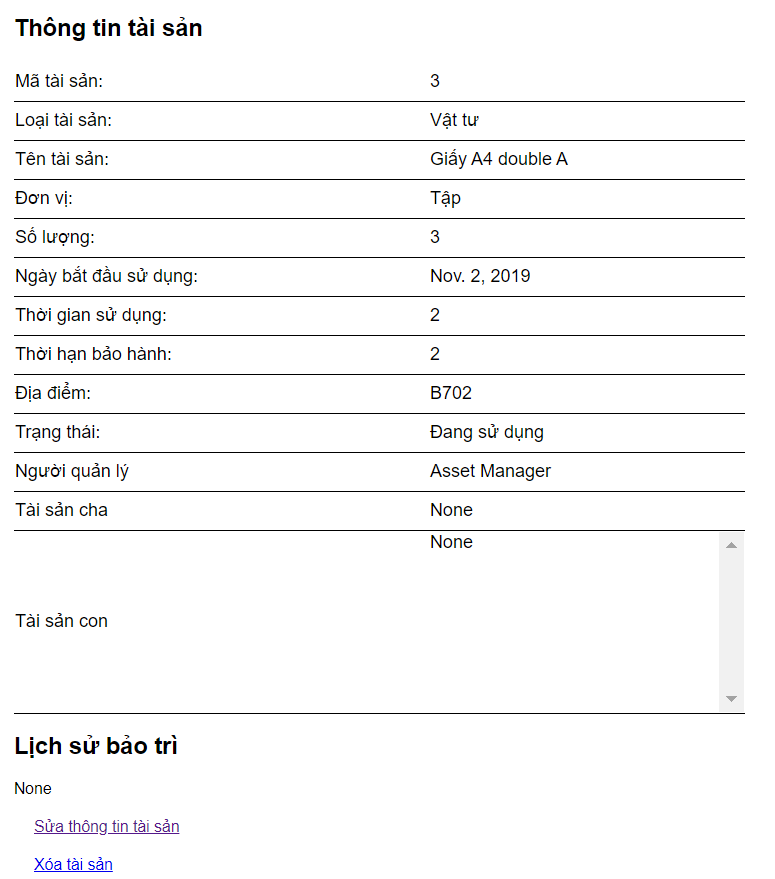
Giao diện tìm kiếm gồm các trường thông tin cần để lọc dữ liệu như ID tài sản, loại tài sản, tên, thời gian bắt đầu sử dụng, hiện trạng. Trường ID và tên tài sản có định dạng là text field vì các giá trị trong trường này có thể là các đoạn text bất kì. Trường loại tài sản và hiện trạng là drop list vì giá trị trong các trường này là các giá trị cho trước. Cuối cùng, trường ngày tháng có định dạng date để hạn chế nhập sai dữ liệu từ người dùng.

*Giao diện danh sách kết quả tìm kiếm tài sản:*



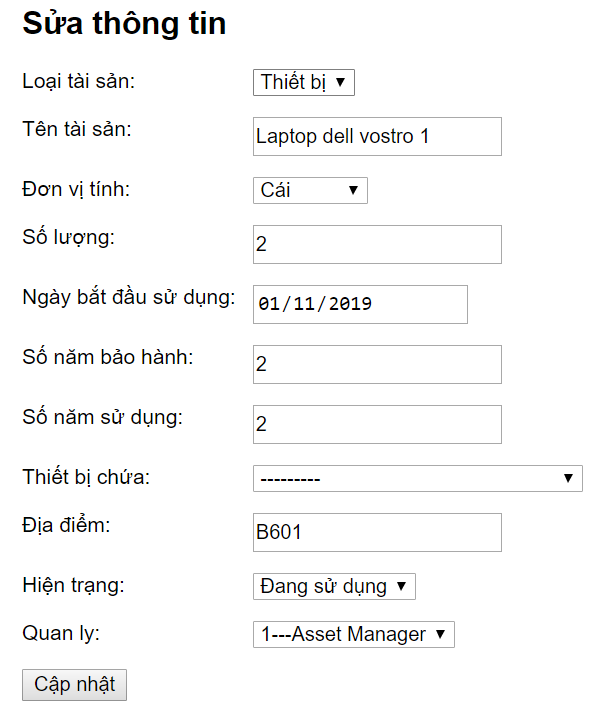
Giao diện danh dách tài sản được thiết kế là 1 bảng thông tin với các trường dữ liệu mã tài sản, loại tài sản, tên tài sản, ngày bắt đầu sử dụng, số năm sử dụng. Tên tài sản cũng là 1 đường dẫn tới giao diện chi tiết của tài sản đó để thực hiện chức năng xem chi tiết tài sản sau khi người dùng tìm kiếm.

*Giao diện chi tiết tài sản:*



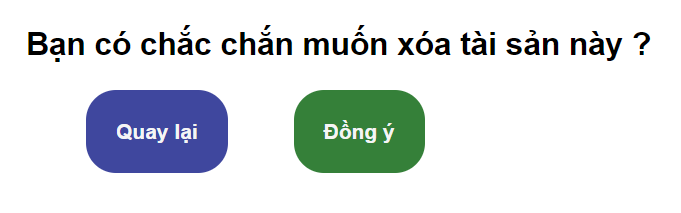
Giao diện chi tiết tài sản là 1 bảng thông tin chứa các thông tin về vật tư thiết bị được lưu trữ trong cơ sở dữ liệu như mã tài sản, loại tài sản, tên tài sản, số lượng, người quản lý, lịch sử bảo trì …. Ngoài ra giao diện còn có các đường dẫn tới giao diện sửa thông tin tài sản và giao diện xóa tài sản.

*Giao diện sửa thông tin tài sản:*



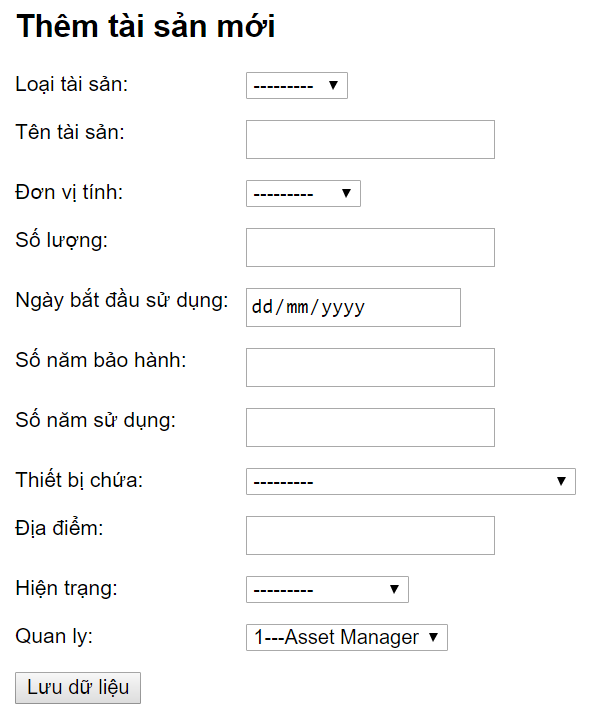
Giao diện sửa thông tin tài sản được thiết kế là 1 biểu mẫu (form) đã điền sẵn các thông tin về tài sản cần thay đổi thông tin. Người dùng chỉ cần thay đổi thông tin trong các trường này rồi bấm nút cập nhật là hệ thống sẽ ghi lại những thay đổi đó vào cơ sở dữ liệu tài sản

*Giao diện xóa tài sản:*



Giao diện xóa tài sản gồm 1 câu hỏi từ hệ thống và 2 nút bấm để người dùng xác nhận xóa tài sản hoặc quay lại giao diện chi tiết tài sản.

*Giao diện thêm tài sản:*



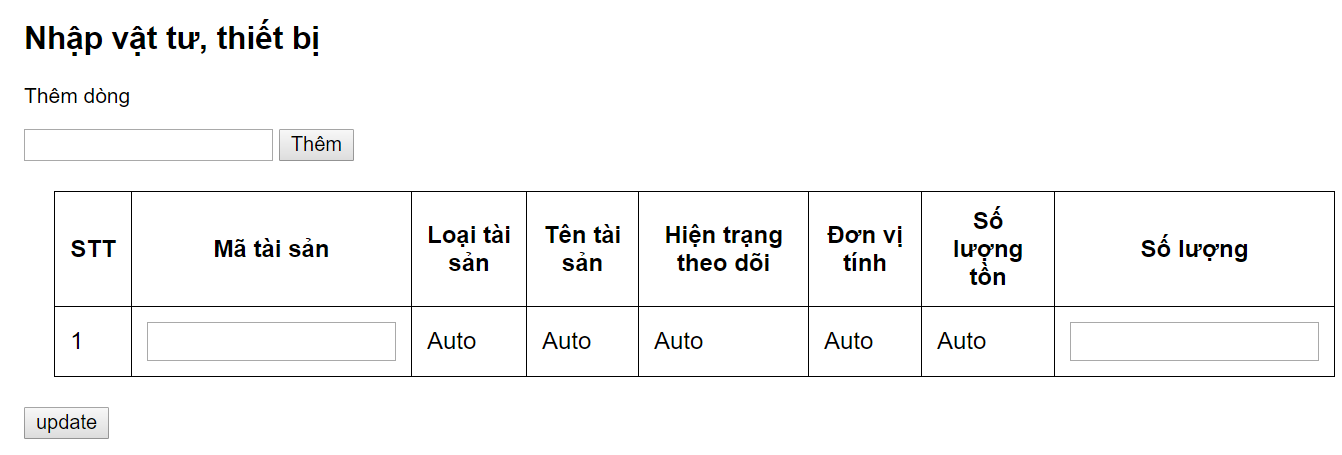
Giao diện thêm tài sản mới là 1 biểu mẫu trắng gồm các thông tin về tài sản cần được lưu vào cơ sở dữ liệu. Các trường Loại tài sản, Thiết bị chứa, Hiện trạng, Quản lý là nhận các giá trị cho trước nên được thiết kế là droplist, trường ngày bắt đầu sử dụng là cần có định dạng ngày tháng. Các trường còn lại nhận giá trị text. Sau khi điền các thông tin, người dùng sẽ click nút lưu dữ liệu để thêm tài sản.

*Giao diện thêm bảo trì:*



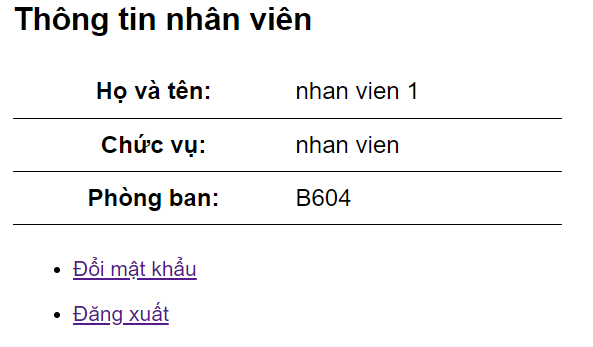
Giao diện thêm thông tin bảo trì mới được thiết kế tương tự như giao diện thêm tài sản mới. Trường tài sản bảo trì nhận giá trị là 1 tài sản cho trước nên được thiết kế là droplist, trường ngày bảo trì có định dạng ngày tháng. Các trường tiêu đề, nội dung là văn bản nên có định dạng text.

*Các giao diện cập nhật số lượng tài sản: nhập kho, xuất kho, kiểm kê:*



Các giao diện nhập, xuất, kiểm kê vật tư thiết bị được thiết kế là 1 biểu mẫu gồm 2 phần. Phần 1 là 1 biểu mẫu dùng để điền thông tin về mã tài sản, và số lượng cần cập nhật với tài sản đó. Các trường loại tài sản, tên tài sản, đơn vị tính, số lượng tồn sẽ được tự động điền khi trường mã tài sản được điền. Sau khi điền các thông tin về tài sản cập nhật, người dùng chọn nút update để cập nhật thông tin vào cơ sở dữ liệu. Phần thứ 2 của giao diện là biểu mẫu để thêm số dòng tương ứng với số tài sản cần cập nhật số liệu vào biểu mẫu. Người dùng nhập vào số dòng cần thêm và chọn nút thêm, số dòng tương ứng sẽ được thêm vào biểu mẫu ban đầu.

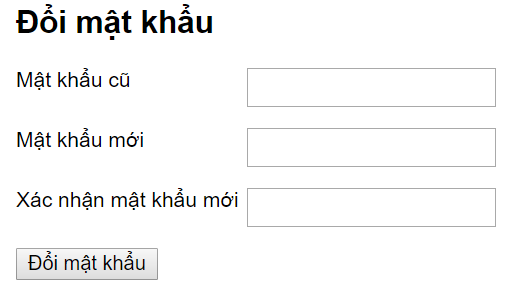
*Giao diện thông tin cá nhân:*



Giao diện thông tin cá nhân được thiết kế là 1 bảng với cột 1 là tên các trường dữ liệu về nhân viên, cột 2 là thông tin về nhân viên đó. Ngoài ra nó còn 2 đường dẫn tới giao diện đổi mật khẩu và giao diện đăng xuất.

*Giao diện đổi mật khẩu:*

Giao diện đổi mật khẩu được thiết kế là 1 biểu mẫu trắng với các trường cần điền là mật khẩu hiện tại, mật khẩu mới và xác nhận mật khẩu

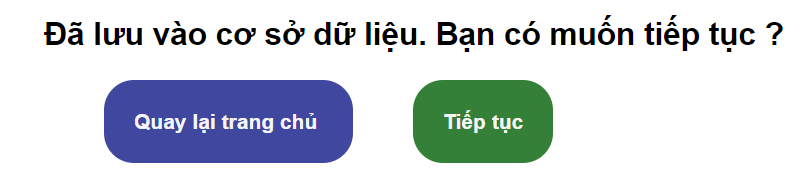


*Giao diện đăng nhập đăng xuất:*



Giao diện đăng nhập/đăng xuất là 1 biểu mẫu trắng với 2 trường user và password để người dùng đăng nhập vào hệ thống.

*Giao diện hệ thống thông báo hoàn thành 1 chức năng:*



Giao diện là 1 thông báo của hệ thống, với 2 nút bấm giúp người dùng quay lại trang chủ hoặc tiếp tục thực hiện chức năng vừa hoàn thành với dữ liệu khác.

# CHƯƠNG IV: CÀI ĐẶT PHẦN MỀM

1. ***Backend***

Phần mềm được xây dựng trên nền tảng web, sử dụng Django framework. Django được thiết kế để xây dựng những phần mềm có những tính chất sau:

***Hoàn thiện:***

Django được thiết kế cung cấp hầu hết mọi công cụ mà lập trình viên cần để phát triển sản phẩm hoàn thiện như module kết nối cơ sở dữ liệu, điền template. Các công cụ này được đóng gói trong 1 sản phầm, hoạt động đồng bộ theo thiết kế kiến trúc chung của toàn hệ thống

***Linh hoạt:***

Django được sử dụng để xây dựng hầu hết mọi loại website ví dụ hệ thống quản lý, mạng xã hội, từ điển…Nó có thể hoạt động với mọi loại framework phía người dùng và truyền tải nội dung với hầu hết mọi định dạng (Html, RSS, Json, XML…).

***Bảo mật:***

Django giúp lập trình viên tránh khỏi rất nhiều những lỗi bảo mật thông thường bằng cách cung cấp 1 bộ khung được thiết kế để bảo vệ website 1 cách tự động. Ví dụ, Django cung cấp cách thức bảo mật để quản lý tài khoản và mật khẩu người dùng, tránh khỏi những lỗi thông thường như đặt thông tin session vào cookie (thay vao đó nó cung cấp 1 mã khóa và dữ liệu thật được lưu trong cơ sở dữ liệu). Framework này cũng bảo vệ website khỏi những rủi ro dễ bị tấn công như SQL injection, cross-site scripting, cross-site request

***Có khả năng mở rộng:***

Django sử dụng kiến trúc component-based share nothing, trong đó các component của hệ thống hoạt động độc lập với nhau và có thể thay thế hoặc thay đổi nếu cần. Với các thành phần độc lập với nhau, hệ thống có thể tăng khả năng truyền tải bằng cách thêm phần cứng ở mọi mức: cache, database và ứng dụng. 1 ví dụ của hệ thống lớn dùng Django là mạng xã hội Instagram.

***Khả năng bảo trì:***

Django sử dụng nguyên lý DRY (don’t repeat yourself) và các mẫu thiết kế để hỗ trợ khả năng tái sử dụng và bảo trì code trong tương lai. Nó cũng được thiết kế để gộp các chức năng thành các ứng dụng có khả năng tái sử dụng cùng với kiến trúc chia module theo mô hình MVC (model view control).

***Đa nền tảng:***

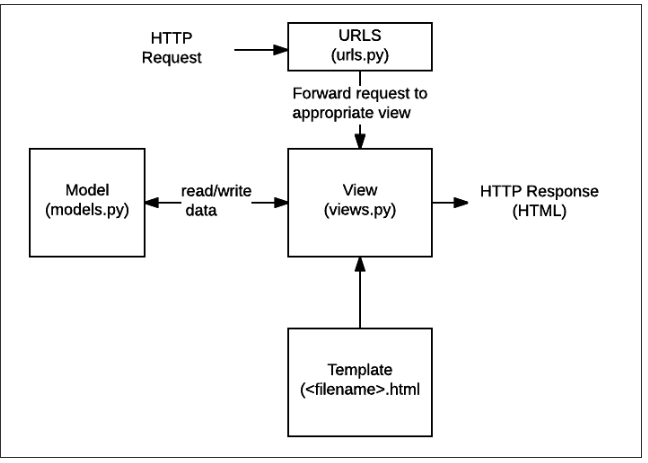
Djan go được viết bằng Python và có thể chạy được trên nhiều nền tảng như MacOS, Linux, Window và được hỗ trợ bởi nhiều nhà cung cấp dịch vụ hosting server.

***Lịch sử phát triển:***

Django được bắt đầu phát triển khoảng từ năm 2003 đến 2005 bởi 1 nhóm lập trình viên phát triển và bảo trì các website tin tức báo chí. Sau khi tạo 1 số websites, các lập trình viên này bắt đầu tái cấu trúc, tái sử dụng các đoạn code và mẫu thiết kế để tạo lên 1 khung chương trình chung, từ đó Django dần được tạo thành. Tới tháng 9 2019, Django 1.0 được phát hành và tới 2017 là Django 2.0. Qua mỗi lần phát hành, các tinh năng mới và lỗi được sửa cung cấp cho Django khả năng kết nối database đa dạng, render các template, tạo lớp các giao diện …

Django được đánh giá là 1 trong những framework phổ biến nhất thế giới dựa trên những thống kê từ website Stackoverflow và Github. Các website nổi tiếng sử dụng framework này bao gồm: Instagram, Mozilla, Pinterest,…

***Các module chính:***



***URLs:*** có tác dụng ánh xạ một HTTP request tới view tương ứng để xử lý yêu cầu dựa vào địa chỉ URL của nó. Nó cũng truyền các dữ liệu số hoặc xâu kí tự trong url tới hàm tạo view tương ứng.

***View:*** là 1 module xử lý yêu cầu từ người dùng, nhận vào yêu cầu HTTP và trả về HTTP response. Mỗi hàm trong module xử lý 1 mẫu url. Nó có thể yêu cầu truy cập databse để đọc hoặc ghi dữ liệu sau đó render các file template để trả về các view cho người dùng

***Models:*** là các đối tượng trong Python định nghĩa cấu trúc dữ liệu của ứng dụng. Nó cũng cung cấp cơ chế truy xuất dữ liệu từ database để phục vụ yêu cầu người dùng.

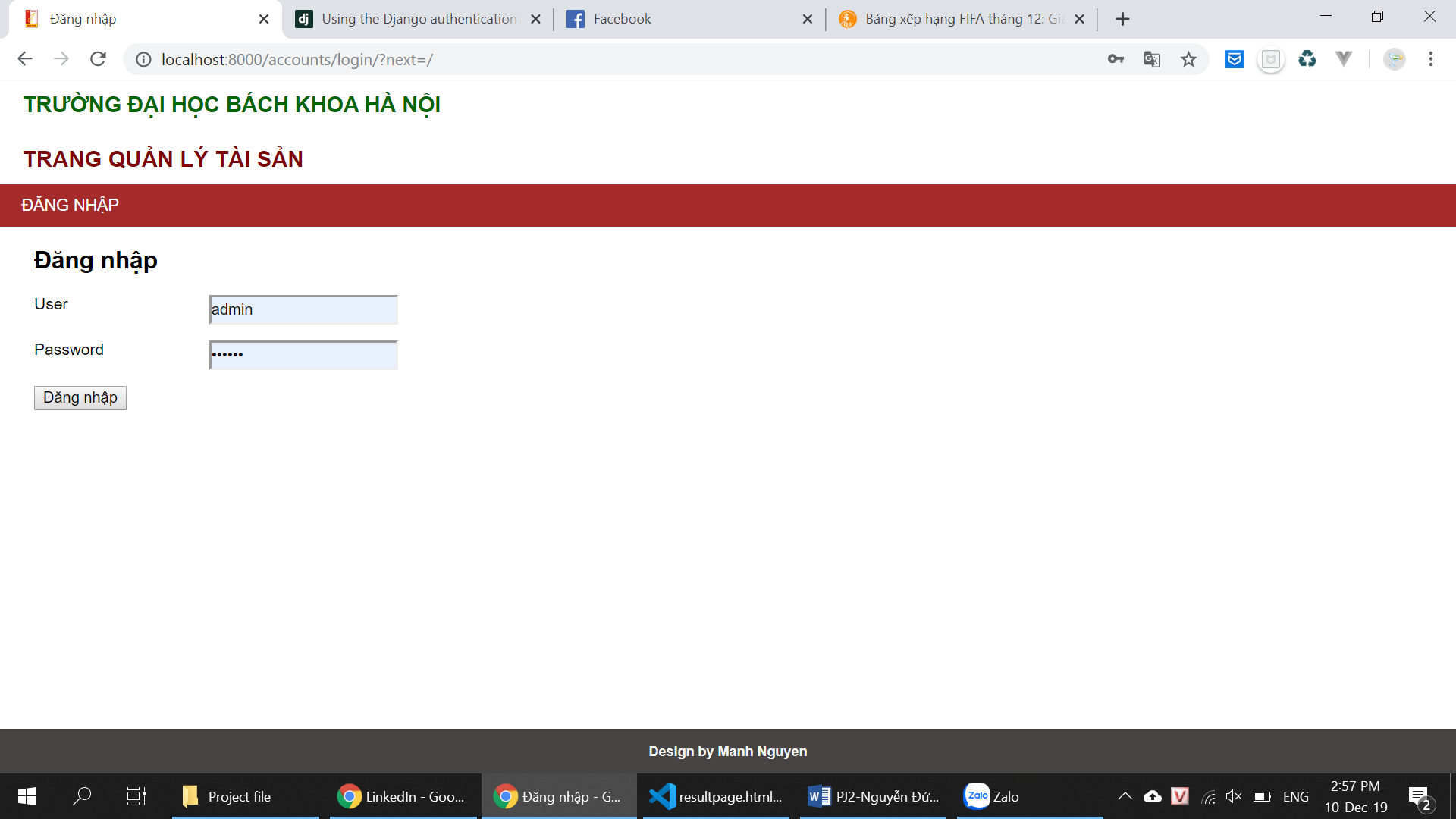
***Templates:*** là 1 file text, sau khi được rennder tạo ra các thành phần giao diện người dùng. Nó chứa các vị trí được định trước để điền thông tin bởi ứng dụng backend.

1. ***Cài đặt thử nghiệm***

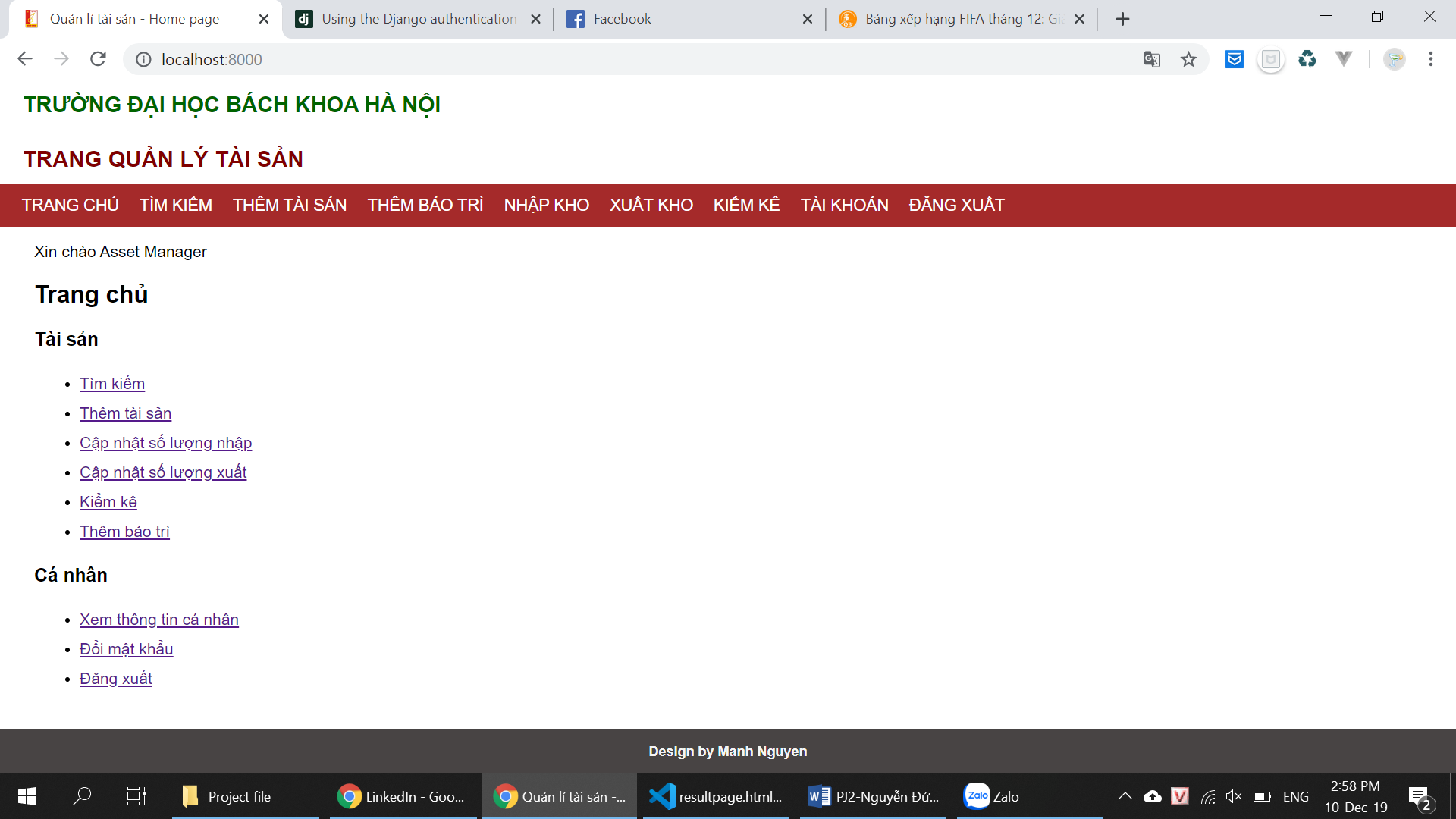
Từ các bước phân tích, thiết kế, em đã tiến hành cài đặt và chạy thử phần mềm và chụp lại được các kết quả như sau:

Chức năng *đăng nhập*

*Thông tin đăng nhập - username: admin, password: 123123*

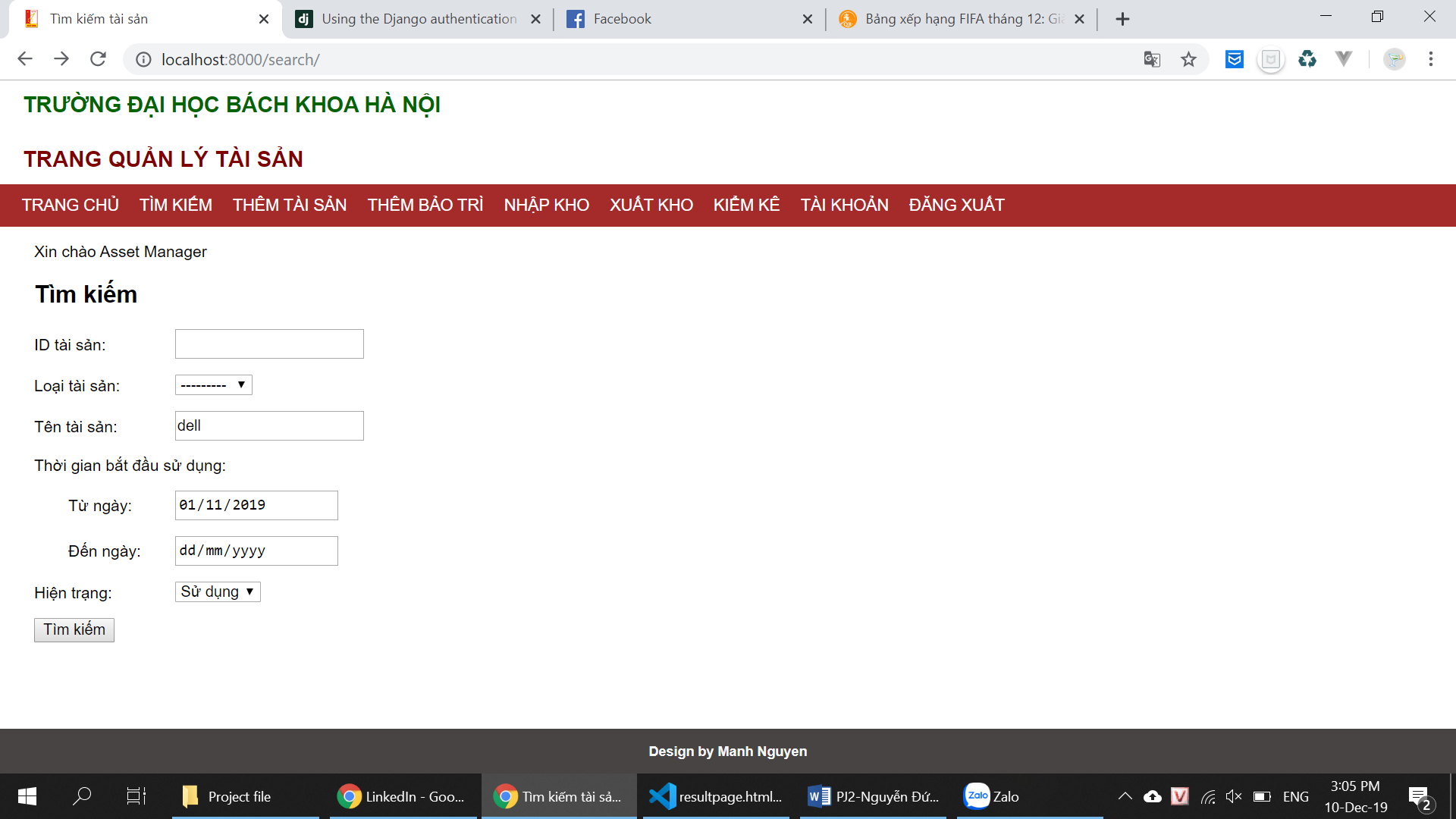


*Trang chủ sau khi đăng nhập:*

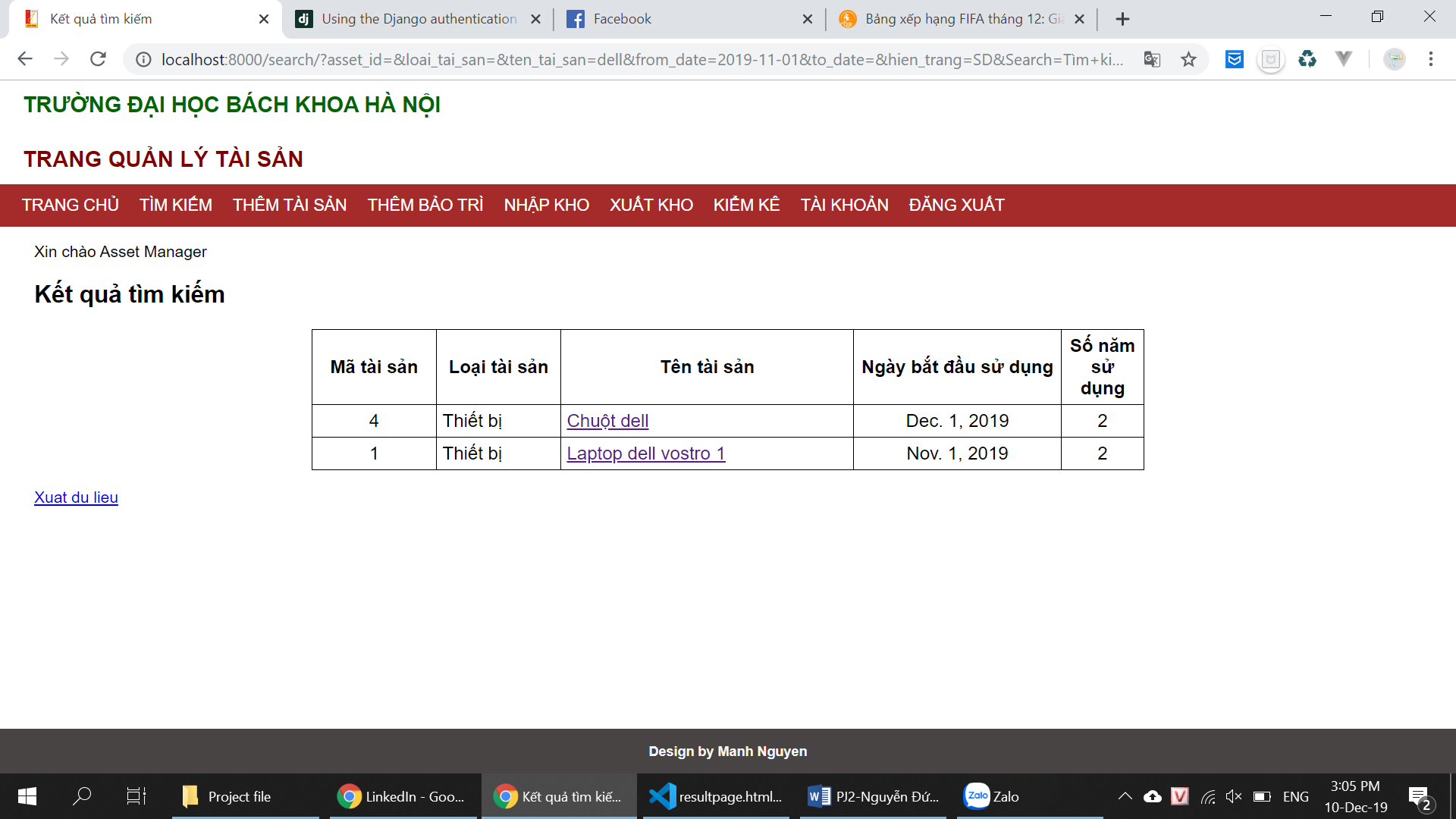


*Chức năng tìm kiếm tài sản:*

Thông tin tìm kiếm: Tài sản có tên chứa xâu kí tự ‘dell’, ngày bắt đầu sử dụng từ sau 1/11/2019 và hiện trạng tài sản là đang sử dụng

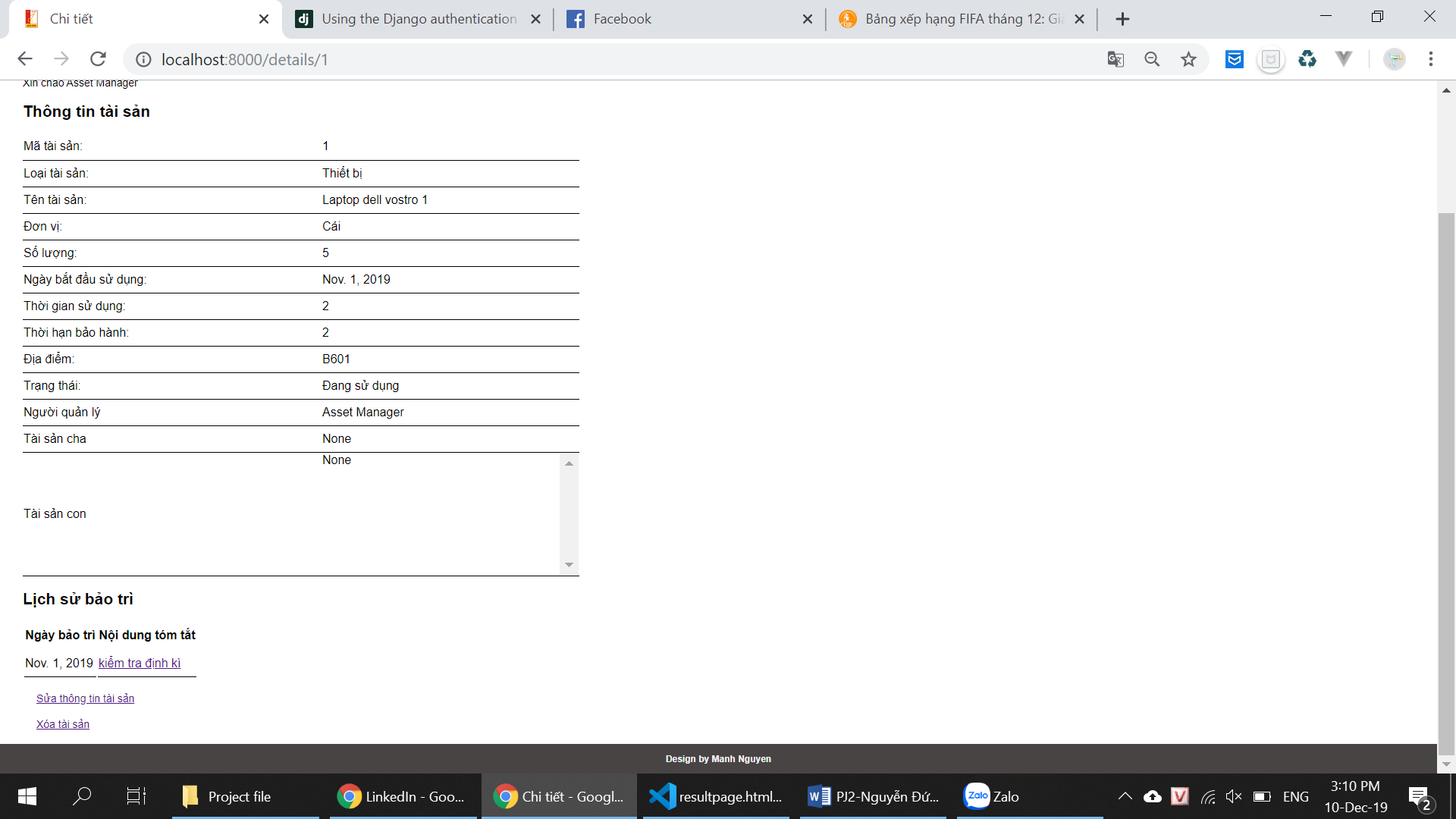


Giao diện kết quả tìm kiếm



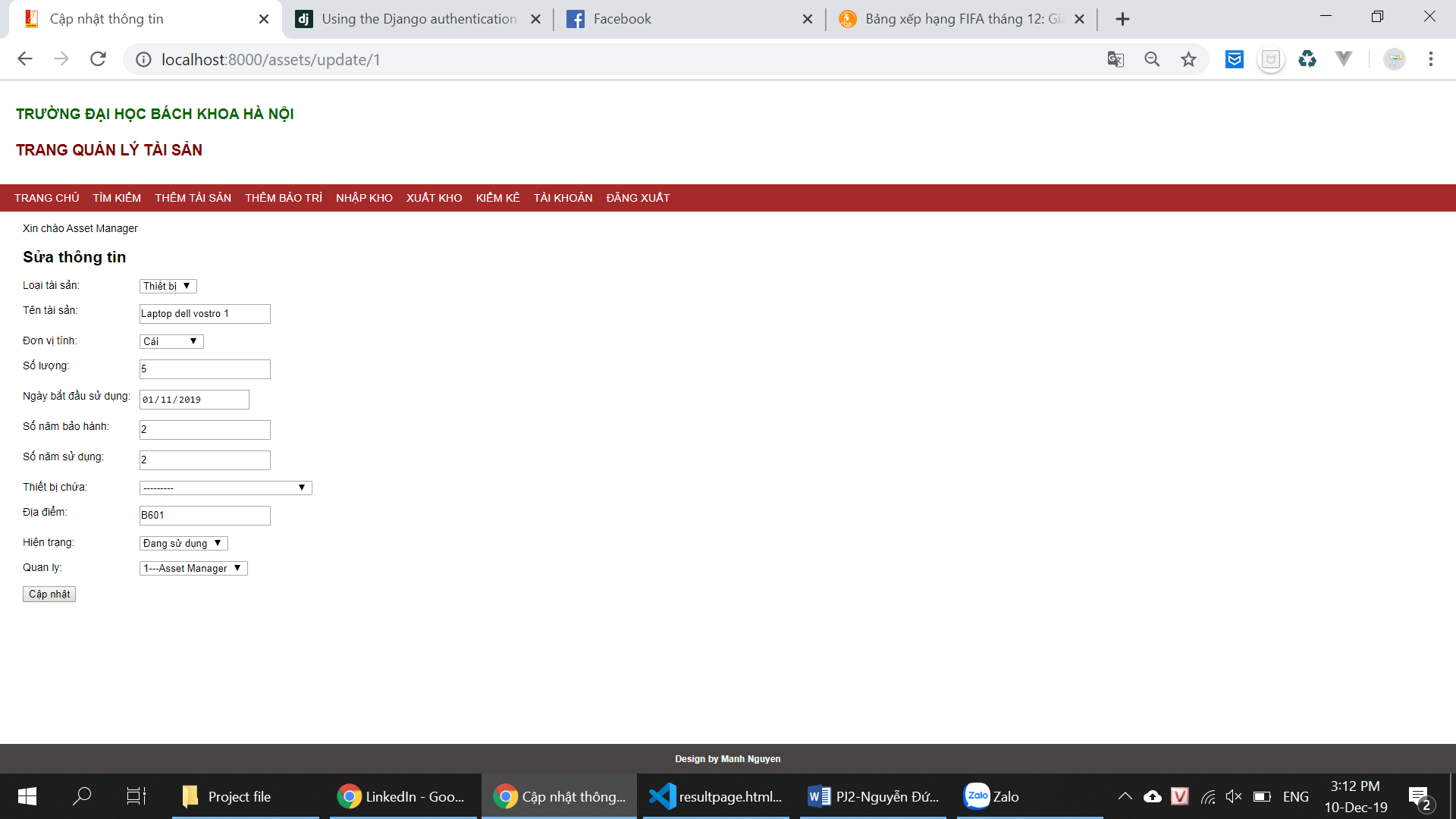
*Chức năng xem chi tiết tài sản mới:*

Từ giao diện tìm kiếm, chọn đường dẫn có tên Laptop Dell Vostro 1 để xem thông tin chi tiết tài sản này kết quả như sau:

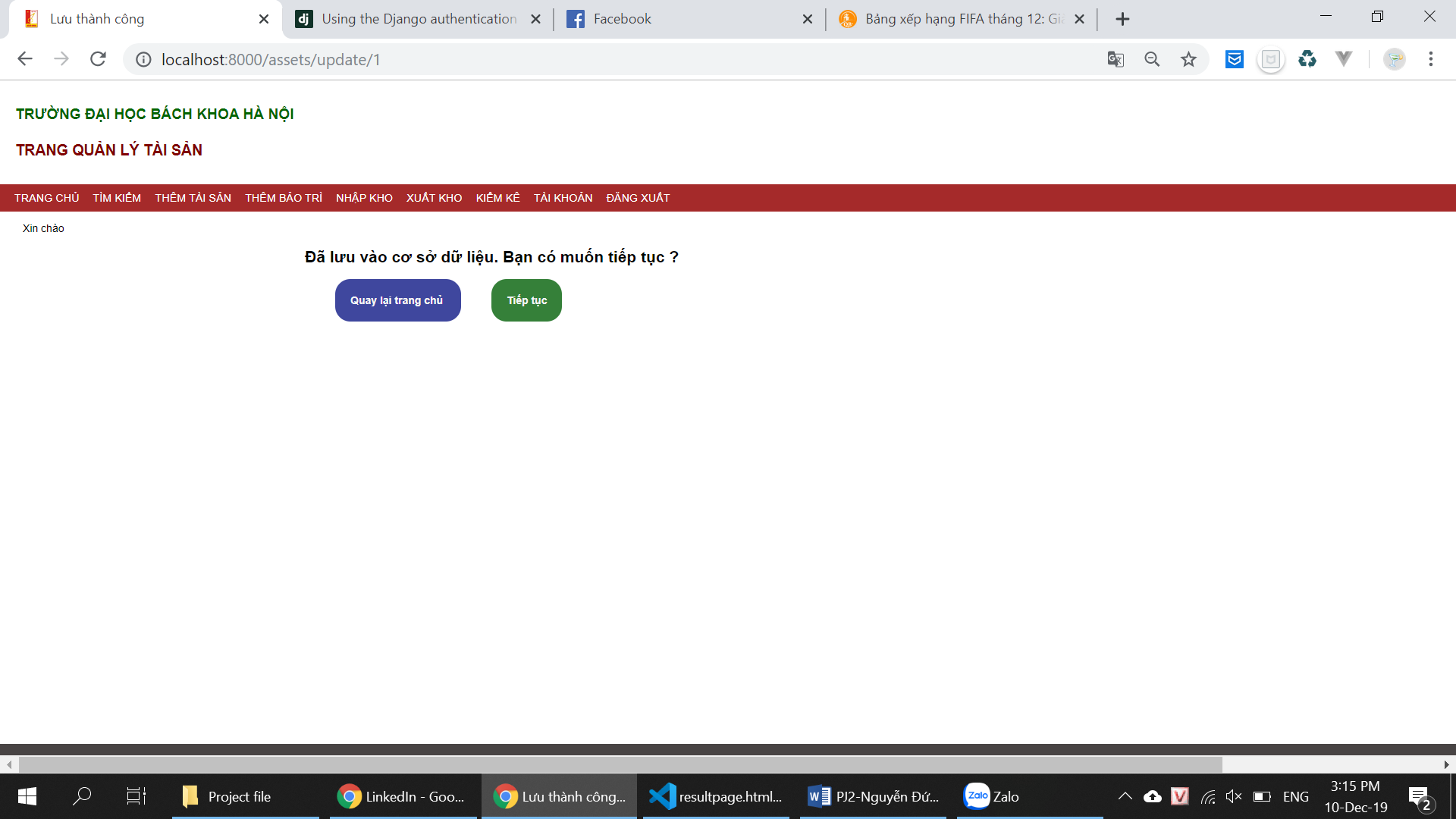


*Chức năng sửa thông tin tài sản:*

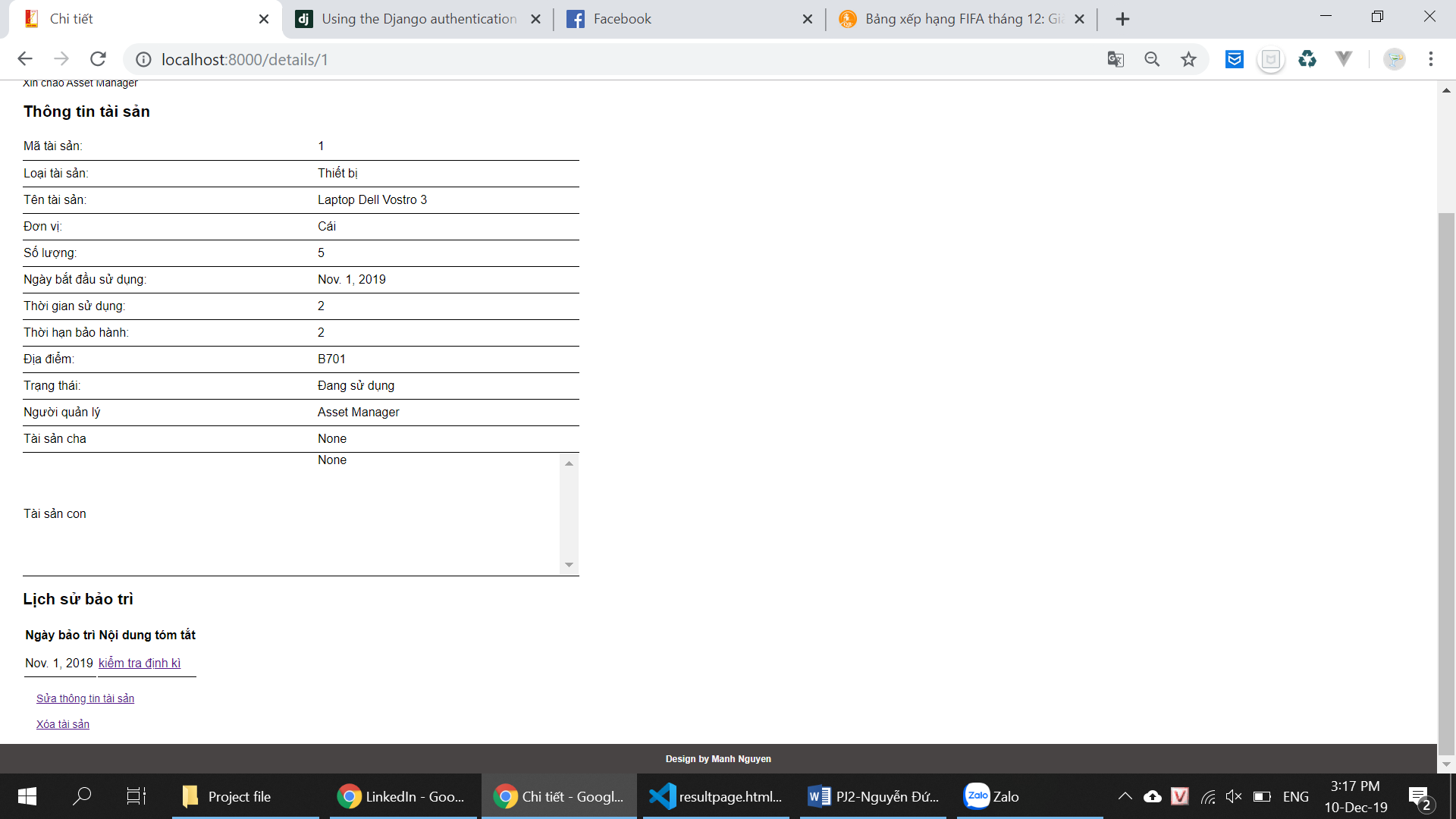
Từ giao diện thông tin chi tiết tài sản Laptop Dell Vostro 1, chọn đường dẫn sửa thông tin tài sản ta được giao diện sau:



Tiến hành sửa tên tài sản từ Laptop dell vostro 1 thành Laptop Dell Vostro 3, địa điểm từ B601 thành B701 và chọn cập nhật, ta được giao diện sau:



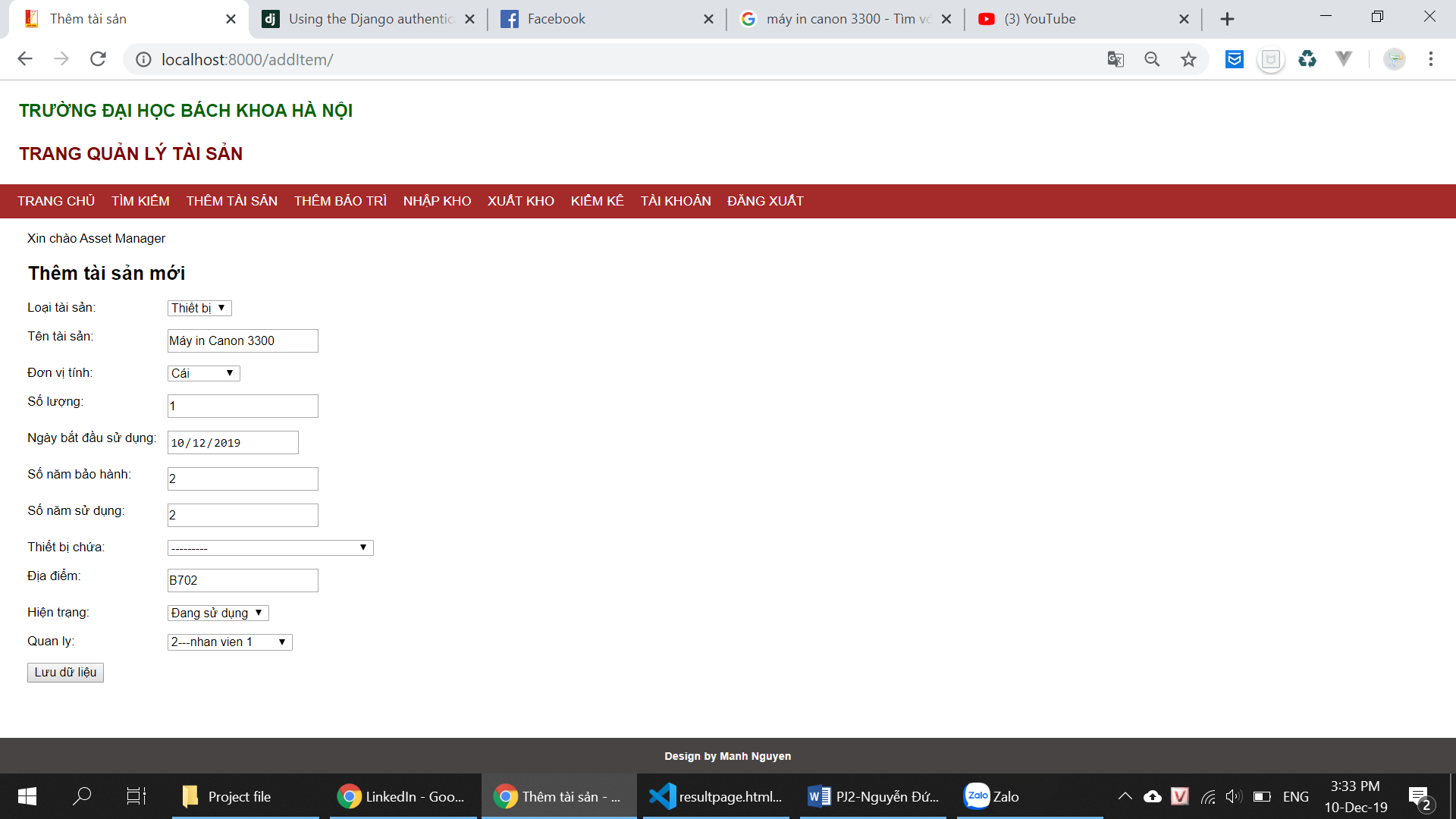
Thông tin mới về tài sản vừa cập nhật như sau:



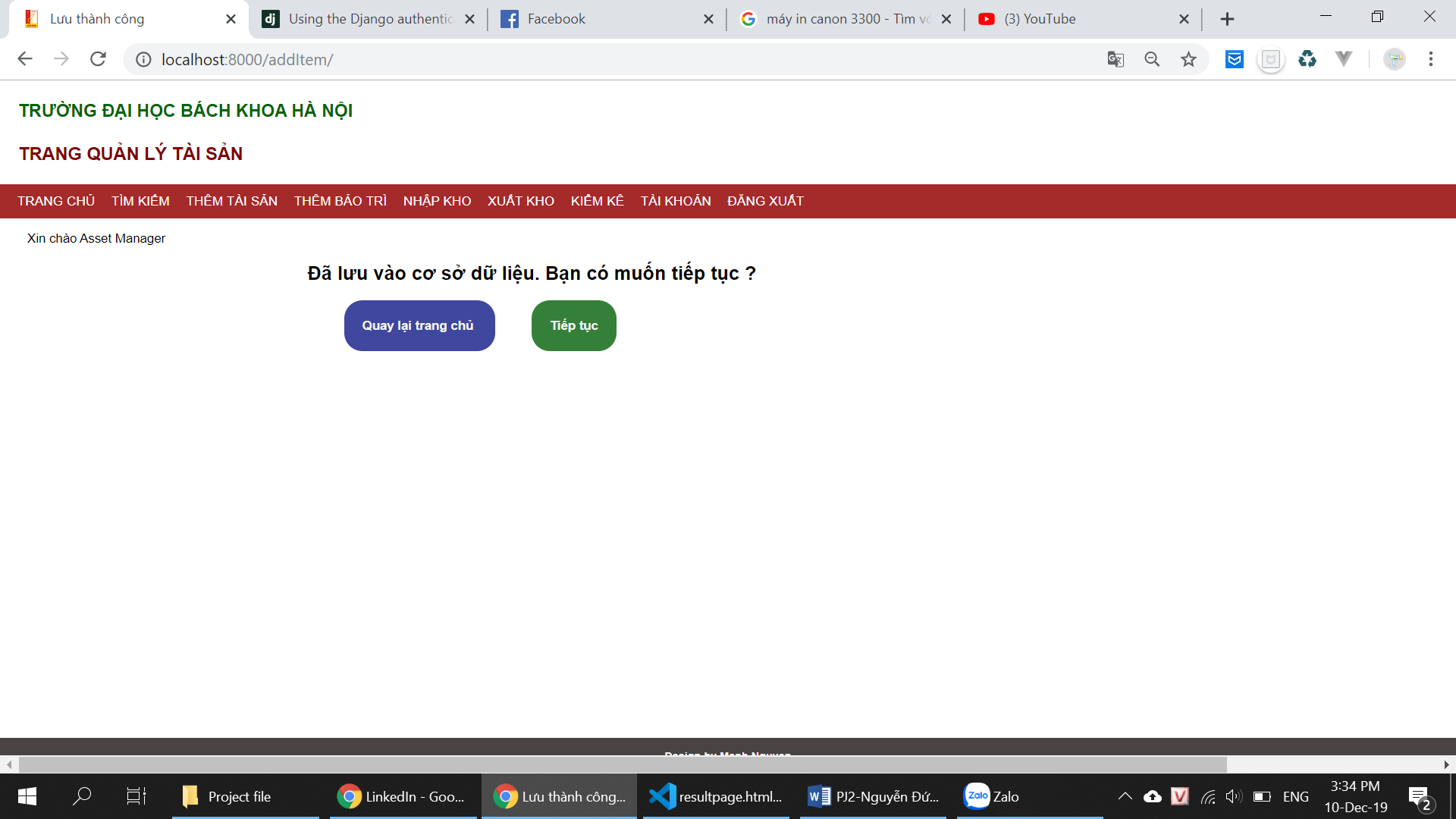
*Thông tin về tên tài sản và địa điểm đặt tài sản đã được thay đổi.*

*Giao diện thêm tài sản mới:*

Thêm 1 tài sản mới là máy in Cannon 3300, loại tài sản thiết bị đơn vị là cái, số lượng 1, ngày bắt đầu sử dụng 10/12/2019, số năm sử dụng 2, số năm bảo hành 2, địa điểm B702, hiện trạng đang sử dụng, và nhân viên quản lý là nhan vien 1. Ta được giao diện sau

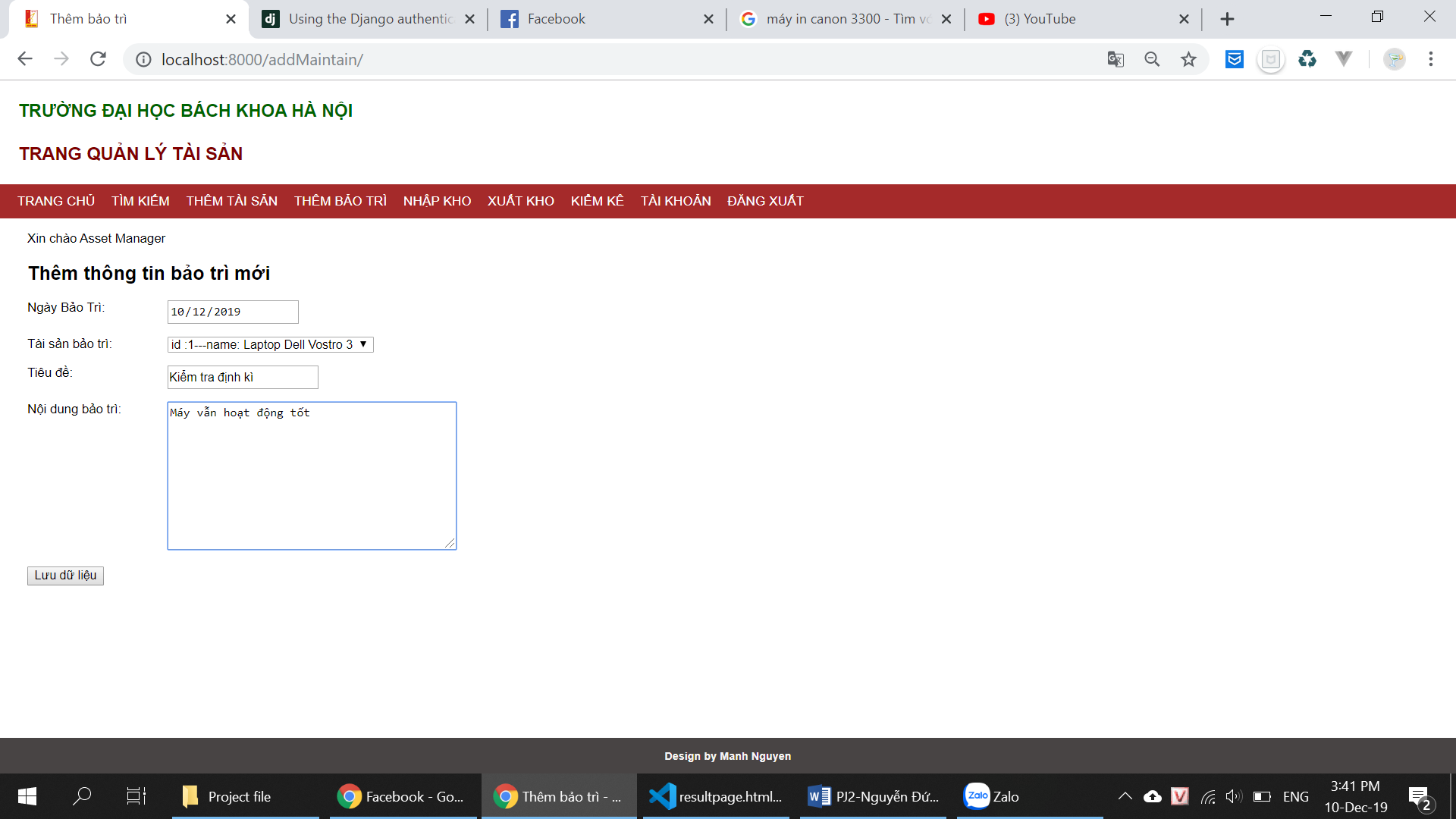


Sau khi chọn lưu dữ liệu ta được giao diện.

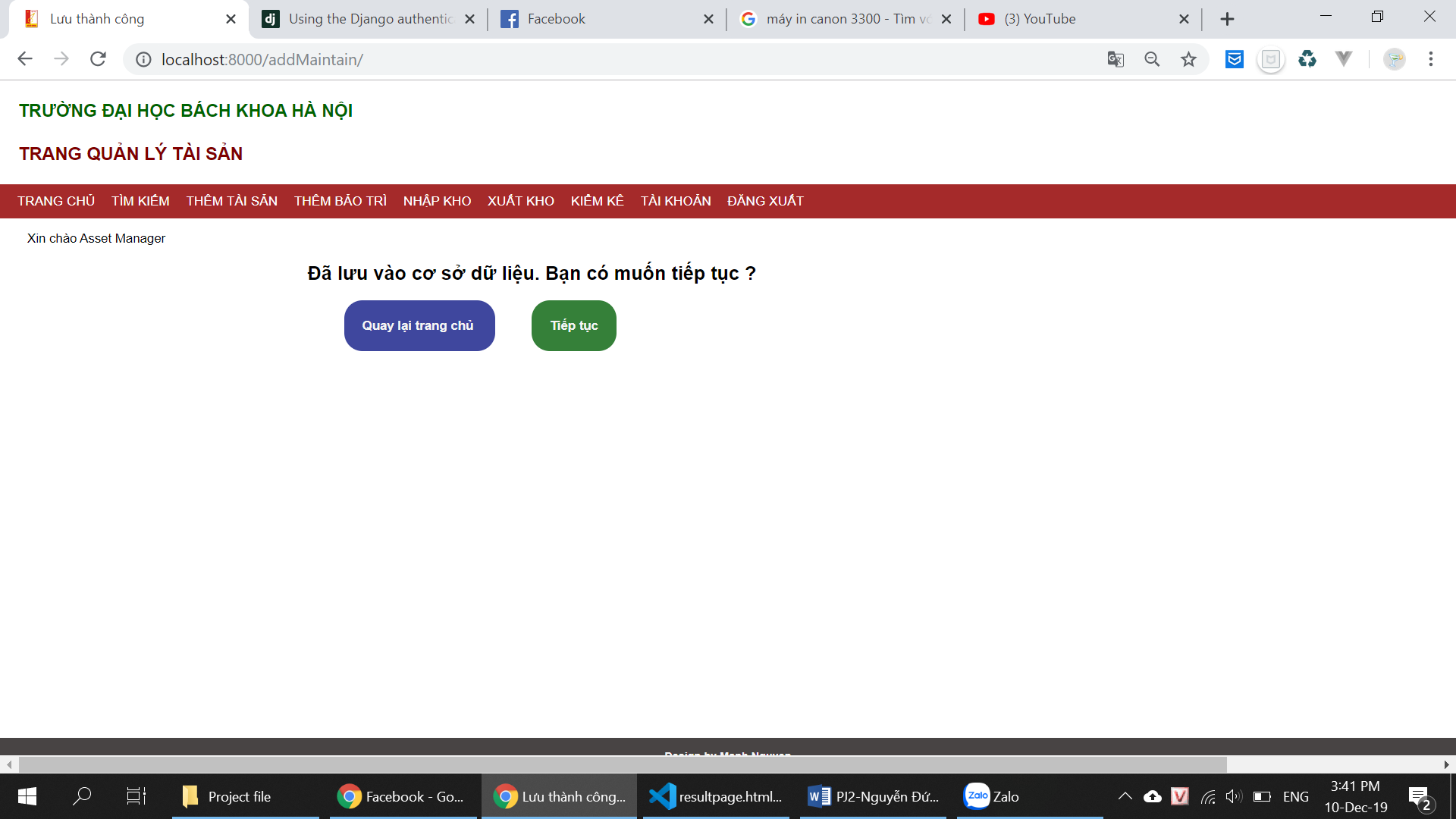


*Chức năng thêm bản ghi thông tin bảo trì cho tài sản:*

Thêm 1 bản ghi bảo trì có nội dung: ngày bảo trì 10/12/2019, Tài sản bảo trì là Laptop Dell Vostro 3 (mã tài sản 1), Tiêu đề kiểm tra định kì, nội dung: máy hoạt động bình thường



Sauk khi chọn lưu, hệ thống trả về thông báo hoàn thành như sau:



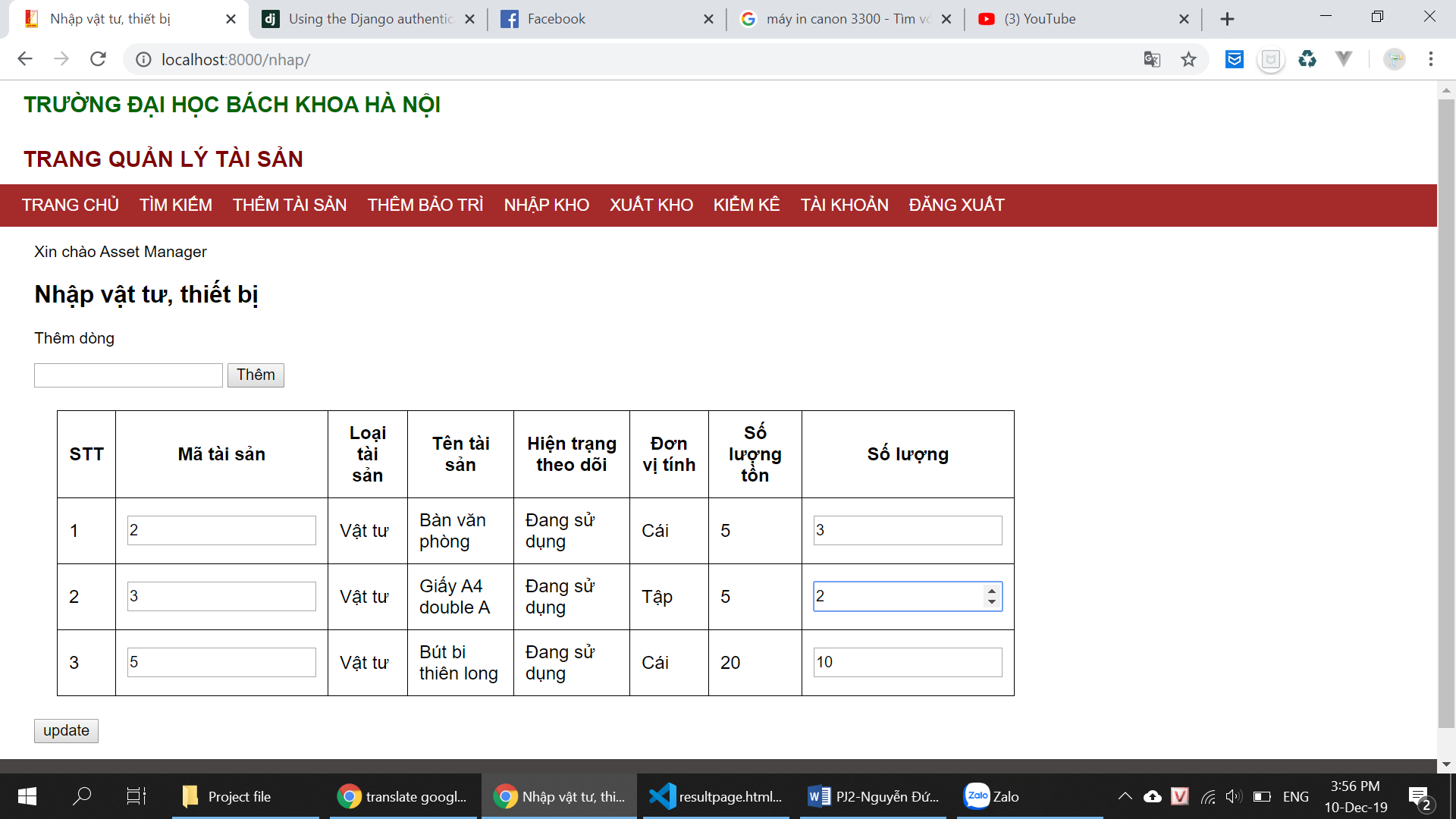
*Chức năng cập nhật nhập vật tư thiết bị:*

Tiến hành nhập vật tư thiết bị với các thông tin sau:

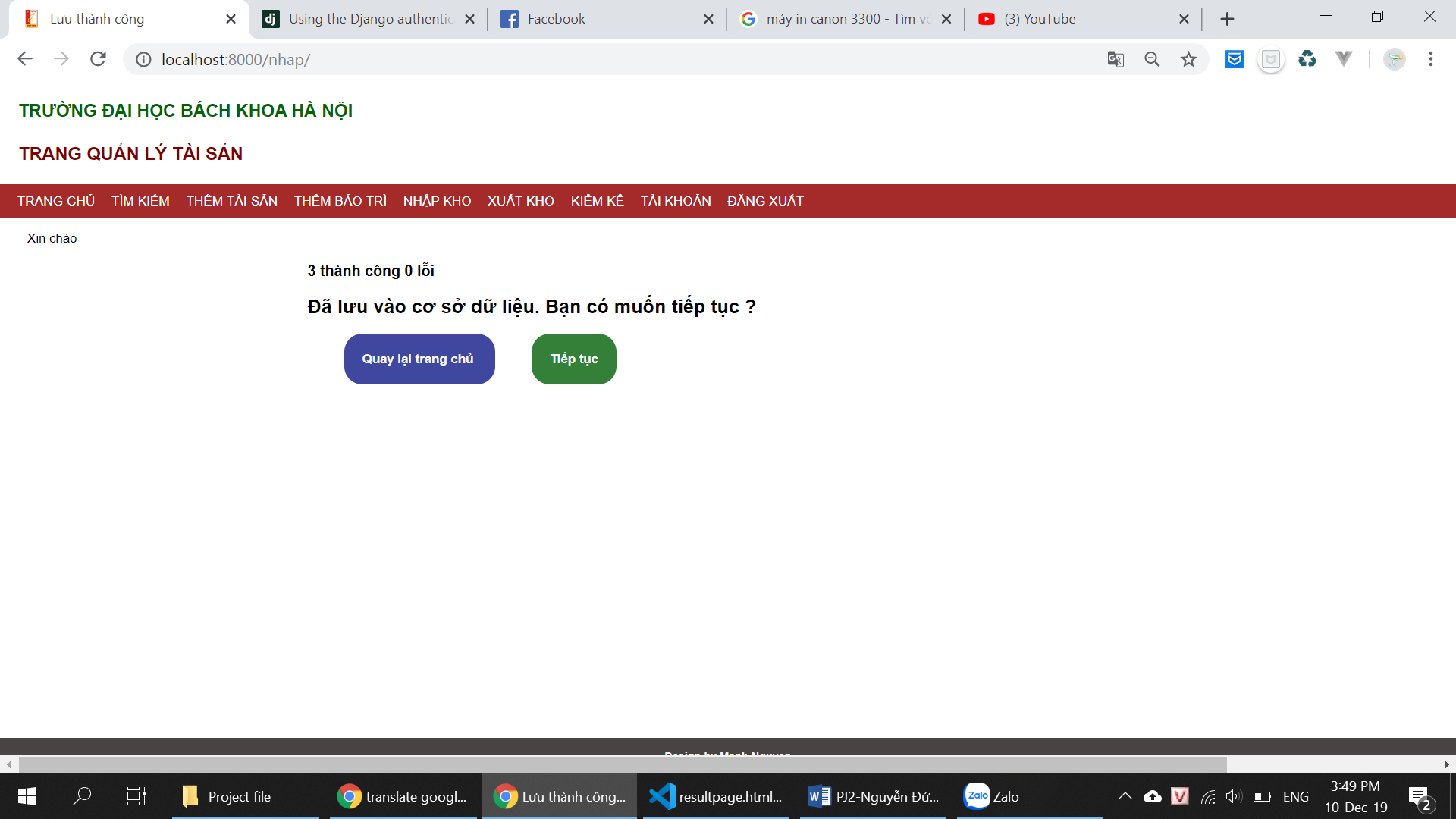
tài sản có mã 2-Bàn văn phòng nhập 3 cái,

tài sản có mã 3-Giấy A4 double A nhập 2 tập

tài sản có mã 5-Bút bi Thiên Long nhập 10 cái

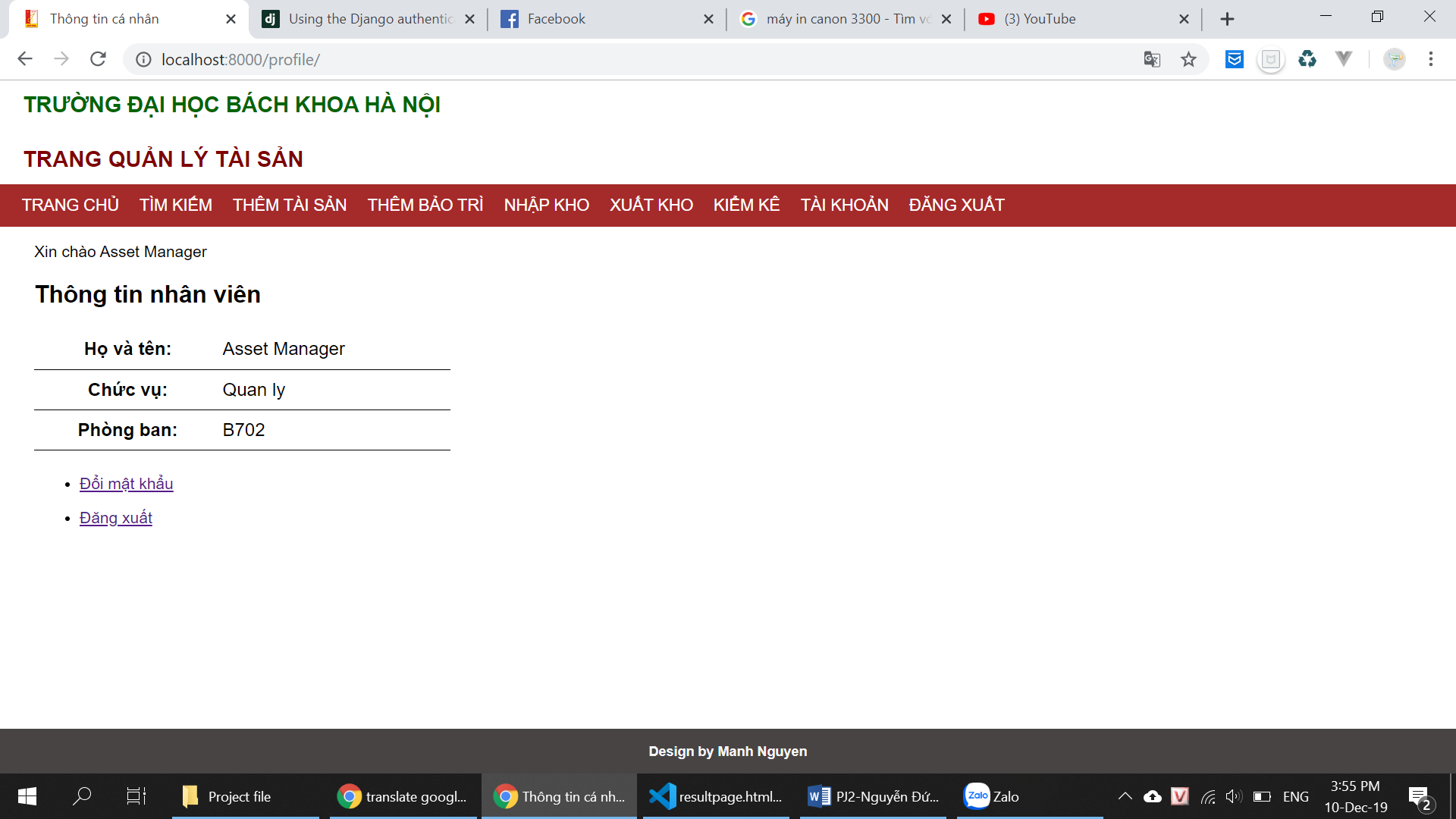


Sau khi chọn cập nhật, hệ thống trả về thông báo sau



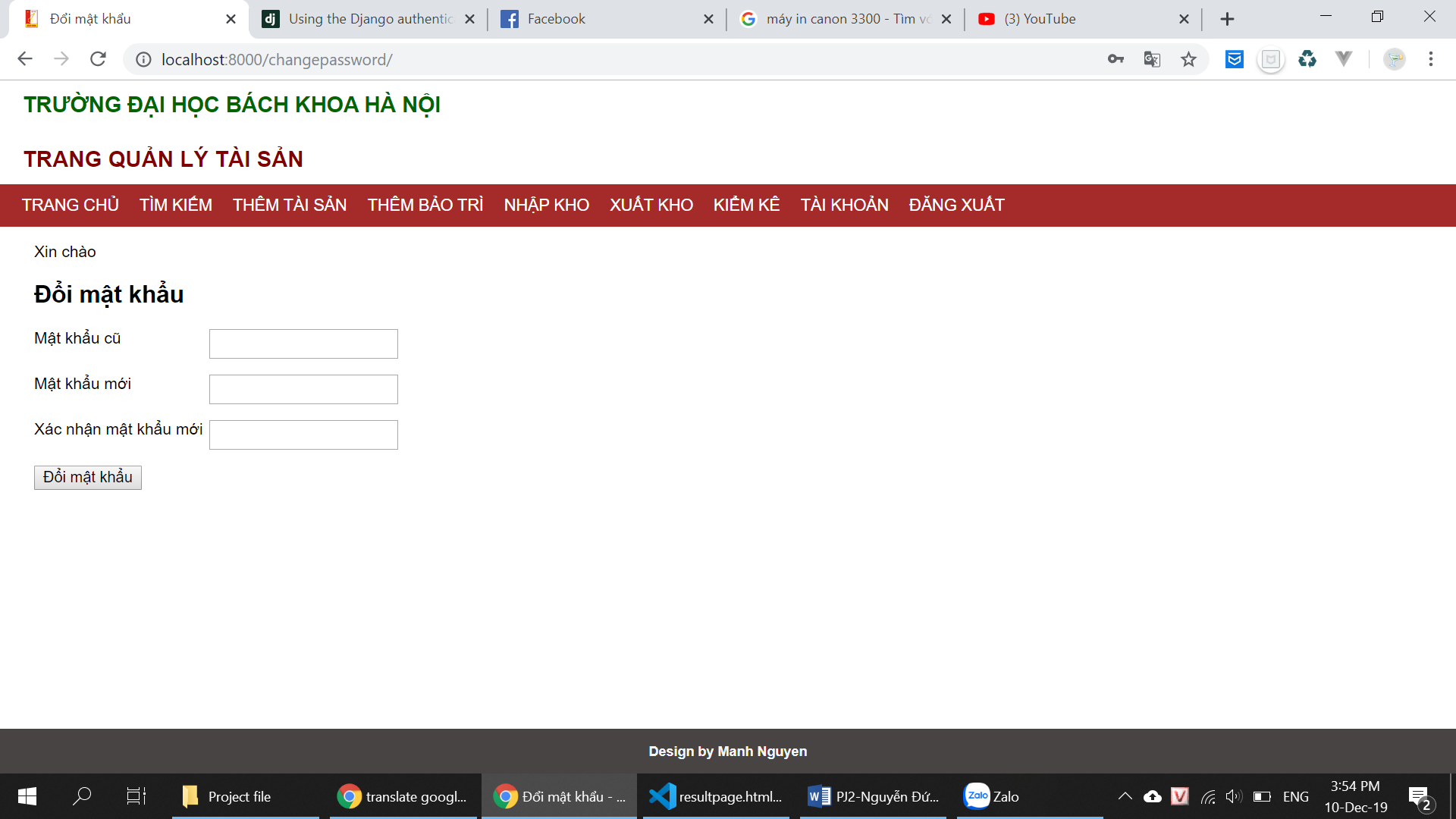
Chức năng xem thông tin cá nhân:

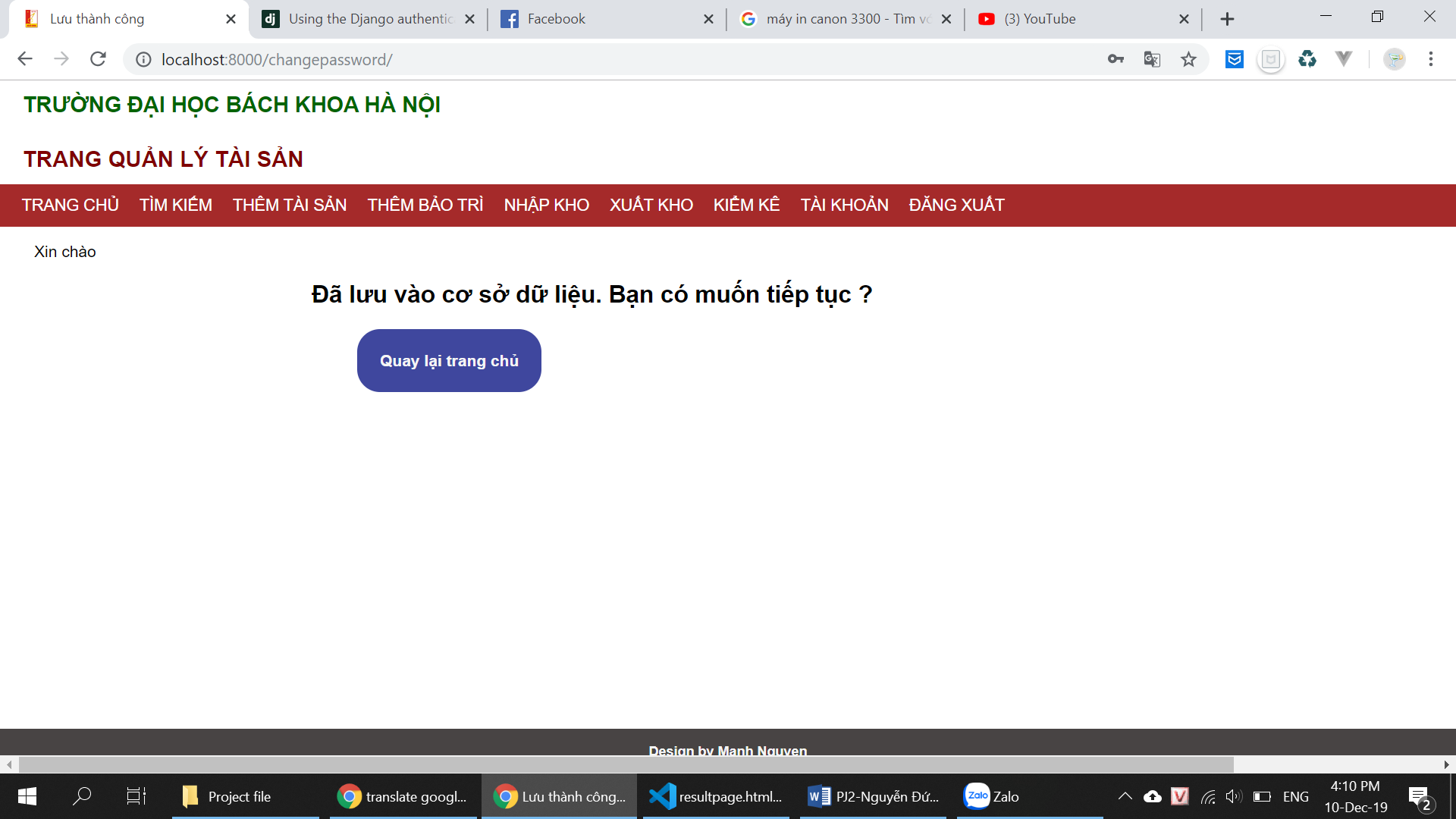
Chọn đường dẫn tài khoản để xem thông tin cá nhân, hệ thống trả về kết quả sau:



*Chức năng đổi mật khẩu:*

Chọn đường dẫn ‘Đổi mật khẩu’ từ giao diện ‘thông tin nhân viên’, điền thông tin về mật khẩu cũ, mới chọn cập nhật. Hệ thống trả về các giao diện sau:





***Kết luận:*** các chức năng của website hoạt động đúng như thiết kế ban đầu của hệ thống.

# KẾT LUẬN

Kết thúc học phần Project 2, em đã học và ứng dụng được những kiến thức sau vào sản phẩm cuối cùng của mình:

* Phân tích yêu cầu người dùng, phân tích ca sử dụng và mô tả ca sử dụng
* Phân tích luồng dữ liệu tổng quát chương trình
* Phân tích và thiết kế cơ sở dữ liệu dựa trên mô hình quan hệ.
* Tìm hiểu về mô hình lập trình web Model View Control (MVC) và Framework Django viết bằng ngôn ngữ Python.
* Thực hành tạo 1 sản phẩm web, tạo cơ sở dữ liệu ở hệ quản trị PostgreSQL và kết nối với ứng dụng backend.
* Thực hiện thao tác CRUD với cơ sở dữ liệu ở backend để trả về kết quả yêu cầu từ phía người dùng bao gồm dữ liệu html, csv. Ngoài ra em cũng áp dụng các chức năng xác thực và phân quyền người dùng vào sản phẩm của mình.
* Thiêt kế 1 bộ giao diện cho website để thực hiện các yêu cầu chức năng. Lập trình giao diện này bằng HTML CSS Javascript đề trả dữ liệu từ backend cho người dùng

Qua học phần này, em đã có cơ hội ứng dụng những kiến thức đã được học vào thực tế, giúp em hiểu sâu hơn những khái niệm và quy trình sản xuất phần mềm.

**Lời cảm ơn:**

Em xin gửi lời cảm ơn chân thành đến ThS Nguyễn Hồng Phương đã hướng dẫn, giải đáp những thắc mắc của em trong suốt quá trình thực hiện học phần này. Em chúc thầy luôn mạnh khỏe và đạt nhiều thành công trong công việc cũng như trong cuộc sống.