**TRƯỜNG ĐẠI HỌC SƯ PHẠM**

**ĐẠI HỌC ĐÀ NẴNG**

**KHOA TIN HỌC**

****

**ĐỒ ÁN MÔN HỌC**

**ĐỀ TÀI:**

**XÂY DỰNG WEBSITE BÁN LAPTOP BẰNG CƠ SỞ DỮ LIỆU TẬP TRUNG**

**Giáo viên hướng dẫn: Võ Công Đình**

**Sinh viên thực hiện: Phan Quang Thắng**

**Mai Đức Tin**

**Phạm Anh**

**Doãn Văn Nam**

**Biền Xuân Thắng**

**Nguyễn Thị Hoàng Anh**

**Lê Thị Trang**

**MỤC LỤC**

[BẢNG TIẾN ĐỘ 2](#_Toc104586756)

[BẢNG PHÂN CÔNG CÔNG VIỆC 2](#_Toc104586757)

[MỞ ĐẦU 3](#_Toc104586758)

[CHƯƠNG 1: CƠ SỞ DỮ LIỆU TẬP TRUNG 4](#_Toc104586759)

[**1.** **Cách cài đặt cơ sở dữ liệu** 4](#_Toc104586760)

[**2.** **Các bảng trong cơ sở dữ liệu** 10](#_Toc104586761)

[**3.** **Mối quan hệ giữa các bảng** 18](#_Toc104586762)

[**4.** **Dữ liệu api của cơ sở dữ liệu tập trung** 19](#_Toc104586763)

[CHƯƠNG 2: NGÔN NGỮ HỔ TRỢ 21](#_Toc104586764)

[**1.** **Giới thiệu ngôn ngữ** 21](#_Toc104586765)

[**2.** **Cách viết API** 24](#_Toc104586766)

[CHƯƠNG 3: PHÂN TÍCH THIẾT KẾ HỆ THỐNG WEBSITE 26](#_Toc104586767)

[**1.** **Giới thiệu về website** 26](#_Toc104586768)

[**2.** **Phân tích thiết kế hệ thống** 26](#_Toc104586769)

[**3.** **Sản phẩm hoàn thiện** 27](#_Toc104586770)

[KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 33](#_Toc104586771)

[TÀI LIỆU THAM KHẢO 34](#_Toc104586772)

# 

# BẢNG TIẾN ĐỘ

|  |  |  |
| --- | --- | --- |
| **Giai đoạn** | **Thời gian** | **Hoàn thành chức năng** |
| Giai đoạn 1 | 28-03-2022 | Cài app cần thiết cho ứng dụng |
| Giai đoạn 2 | 02-04-2022 | Thiết kế website bán điện thoại |
| Giai đoạn 3 | 09-04-2022 | Thiết kế database |
| Giai đoạn 4 | 16-04-2022 | Tạo API và xây dựng front-end cho các trang |
| Giai đoạn 5 | 23-04-2022 | Tạo Trigger, Procedure, Function |
| Giai đoạn 6 | 07-05-2022 | Xây dựng back-end cho các trang |
| Giai đoạn 7 | 20-05-2022 | Chạy demo website thành công |
| Giai đoạn 8 | 21-05-2022 | Viết báo cáo và slide |

# BẢNG PHÂN CÔNG CÔNG VIỆC

|  |  |
| --- | --- |
| **Thành viên** | **Công việc** |
| Phan Quang Thắng | Phân Chia công việc  Đăng nhập,User và thay đổi thông tin User  Quản lí sản phẩm,quản lí đơn hàng  Procedure,Function,View,Transaction |
| Mai Đức Tin | Giỏ hàng  Đặt hàng  Thiết kế các trang Fontend  Function,Triggers |
| Nguyễn Thị Hoàng Anh | Đăng Kí  Đổi mật khẩu người dùng  Triggers,View,Function  Viết báo cáo |
| Lê Thị Trang | Quản lí hãng(Brand)  Dashboard Admin  Triggers,View,Function  Viết Báo Cáo |
| Biền Xuân Thắng | Trang chủ  Triggers |
| Doãn Văn Nam | Trang giới thiệu  Liên hệ  Triggers |
| Phạm Anh | Tìm kiếm sản phẩm  Chi tiết sản phẩm,Triggers,Procedure |

# MỞ ĐẦU

1. **Lý do chọn cơ sở dữ liệu**

Hiện nay với sự phát triển của công nghệ đồng thời nhiều loại cơ sở dữ liệu được sử dụng để xây dựng các app, các web. Tuy nhiên việc sử dụng cơ sở dữ liệu tập trung để xây dựng web sẽ xử lý một số khuyết điểm như là giảm thông tin trùng lặp, tăng khả năng truy xuất thông tin…

Cơ sở dữ liệu tập trung khá là quen thuộc với mọi người. Có thể vận dụng thực tế từ những bài học đã được học ở trong quá trình học.

1. **Lý do chọn đề tài web**

Chúng ta có thể nói rằng thế ký 21 đã và đang chứng kiến sự phát triển mạnh mẽ của ngành Công nghệ thông tin (CNTT). CNTT đã và đang thay đổi thế giới một cách nhanh chóng và tưng bước kéo nền tri thức của nhân loại xích lại gần nhau hơn. Đó là nhờ việc phát triển hệ thống website trên toàn thế giới.

Ngày nay, Website đã đóng một vai trò quan trọng đối với con người chúng ta từ giải trí cho đến quảng cáo, thương mại, quản lý…Website(thương mại điện tử) sẽ dần thay thế những phương thức kinh doanh cũ trong các doanh nghiệp bởi tính ưu việt mà Website mang lại như: nhanh hơn, rẻ hơn, tiện dụng hơn, hiệu quả hơn và không bị giới hạn không gian và thời gian.

Để góp phần thúc đẩy thương mại điện tử ở Việt Nam, nhóm em đã tìm hiểu và thực hiện đề tài “ Xây dựng website bán laptop”.

1. **Vận dụng**

Đối với cơ sở dữ liệu tập trung sẽ vận dụng từ các kiến thức được học ở trường

Các khái niệm lý thuyết về function, view, procedure, trigger…. Để áp dụng trực tiếp vào web để hiểu rõ hơn về cách xử lý cũng như cách để vận dụng cơ sở dữ liệu vào web.

**CHƯƠNG 1: CƠ SỞ DỮ LIỆU TẬP TRUNG**

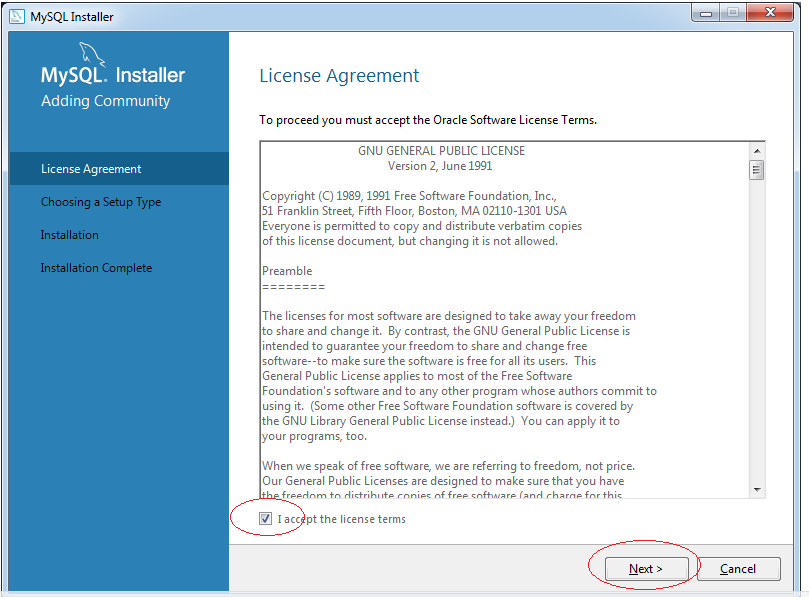
1. **Cách cài đặt cơ sở dữ liệu**

Sử dụng mysql để thực hiện tạo cơ sở dữ liệu cho web. Cách cài đặt như sau:

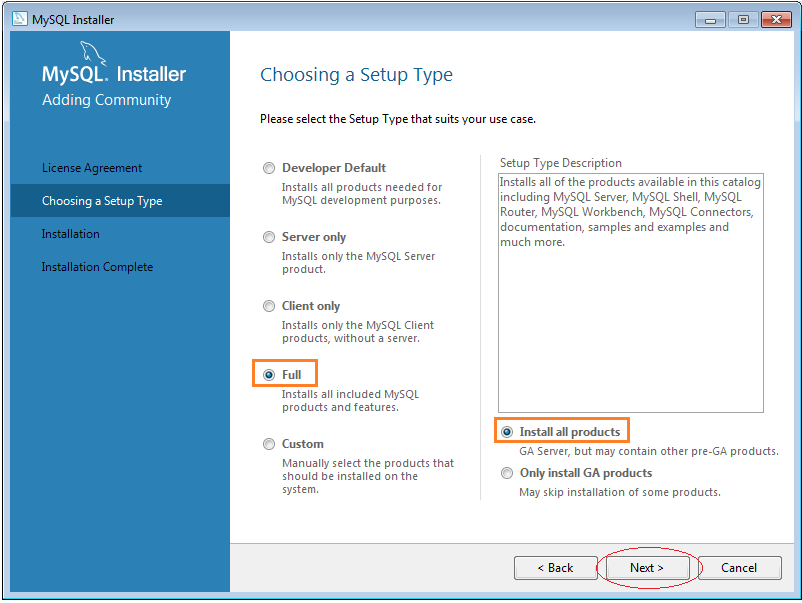
**Bước 1:** Thực hiện download tại link <https://dev.mysql.com/downloads/>

**Bước 2:** Thực hiện cài đặt mysql

Nhấn tick vào **I accept the license terms**, sau đó nhấn **Next**.

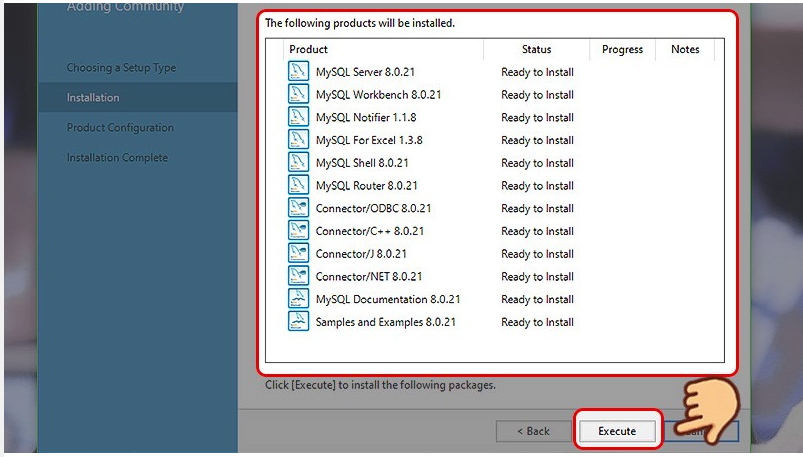


**Bước 3**: Có thể chọn loại cài đặt theo nhu cầu của bạn sử dụng và chọn **next**.



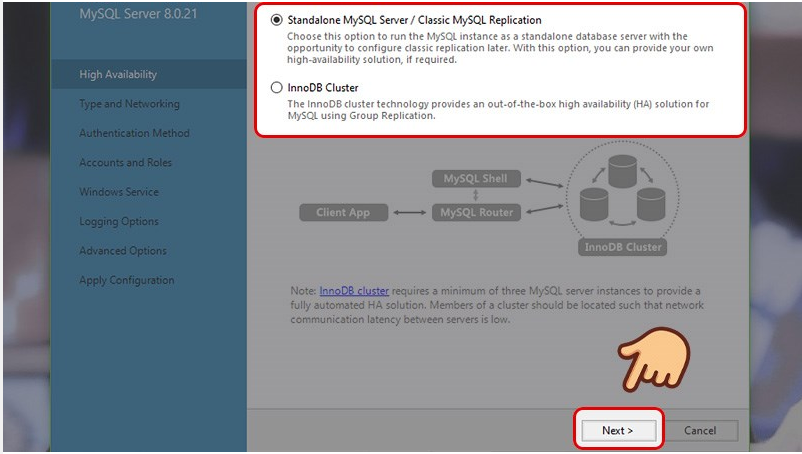
**Bước 4:** Tiếp tục nhấn next

**Bước 5:**  Bộ cài hiển thị danh sách các gói sẽ được cài vào, chọn **Execute**.



**Bước 6:** Đợi một chút để chương trình kiểm tra cài đặt và chọn **Next**.

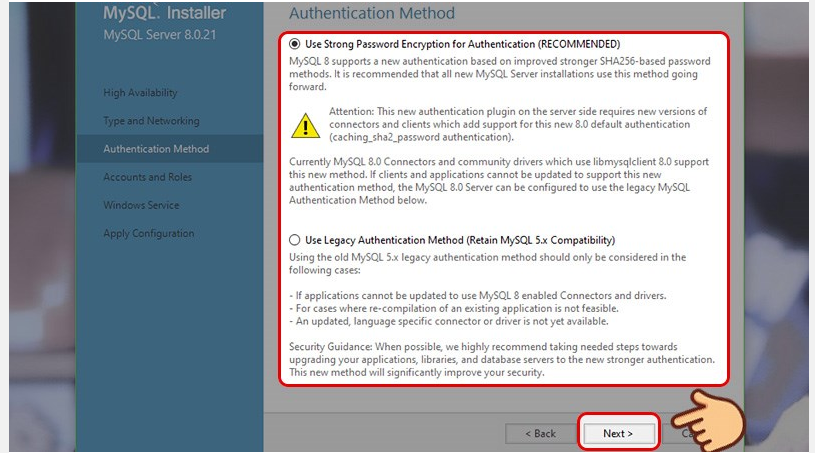
**Bước 7:** Bộ cài đặt tiếp tục tới phần cấu hình MySQL Server, chọn **Next**.

**Bước 8:** Chọn tính khả dụng cho nhu cầu cài đặt, chọn **Next**.

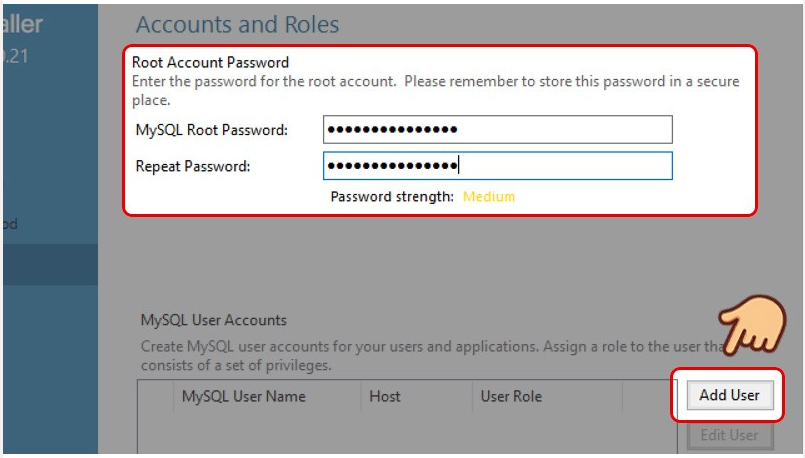
**Bước 9:** Tại bảng cài đặt mạng, hãy chỉnh theo ý muốn hoặc giữ nguyên như cài đặt chuẩn của nhà sản xuất và chọn **Next**.



**Bước 10:** Tại mục **Phương thức chứng thực (Authentication method)** chọn đặt mật khẩu khi sử dụng hay sử dụng chung. Ở đây mình sẽ chọn sử dụng với mật khẩu. Tiếp đến chọn **Next**.

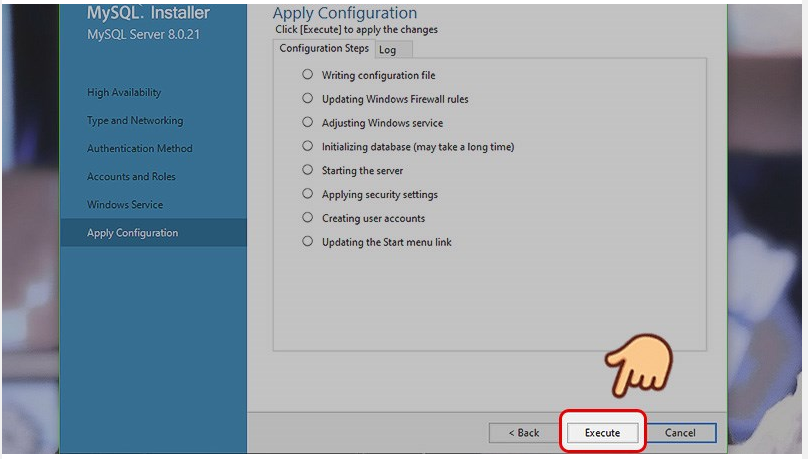


**Bước 11:** Đặt mật khẩu và xác nhận lại mật khẩu của bạn hoặc chọn **Add user** để thêm tài khoản sử dụng.



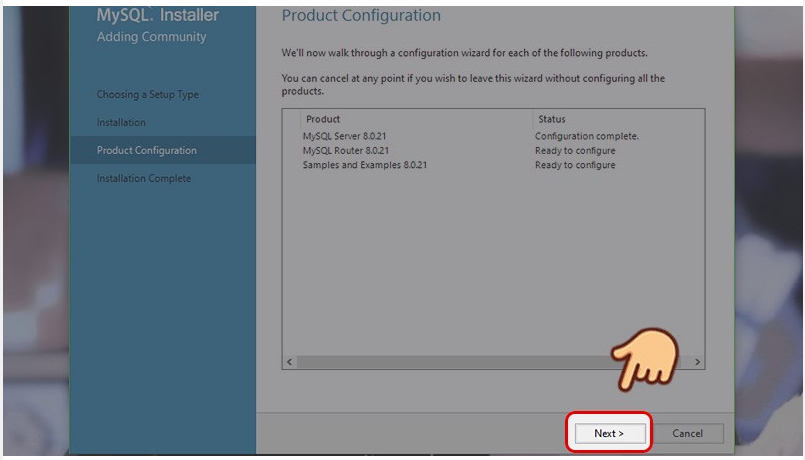
**Bước 12:** Tiếp tục chọn Next.

**Bước 13:** Ở mục **Áp dụng cấu hình (Apply configuration)**, chọn **Execute** để áp dụng.



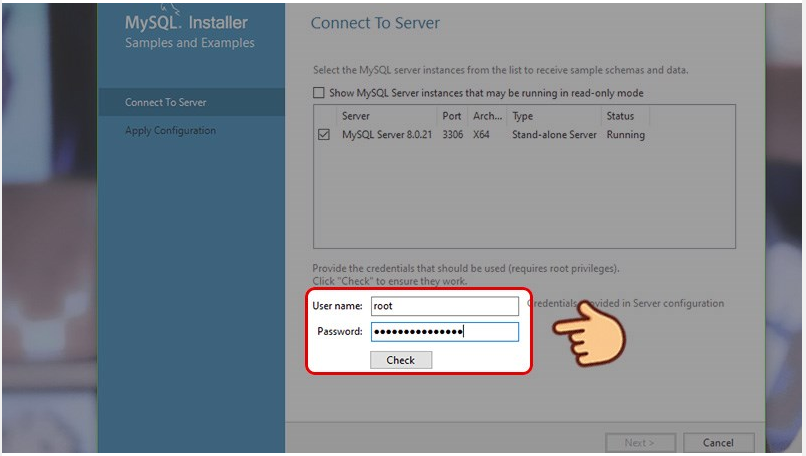
**Bước 14:** Sau đó chọn **Finish** để kết thúc quá trình thiết lập bảo mật.

**Bước 15:** Tại mục **Product Configuration**, chọn **Next**.



**Bước 16:**Sau đó chọn **Finish**.

**Bước 17:** Điền tên, password và chọn **Check**, sau khi check thành công chọn Next.



**Bước 18**: Chọn **Execute** sau đó nhấn  **Finish** để kết thúc quá trình cài đặt

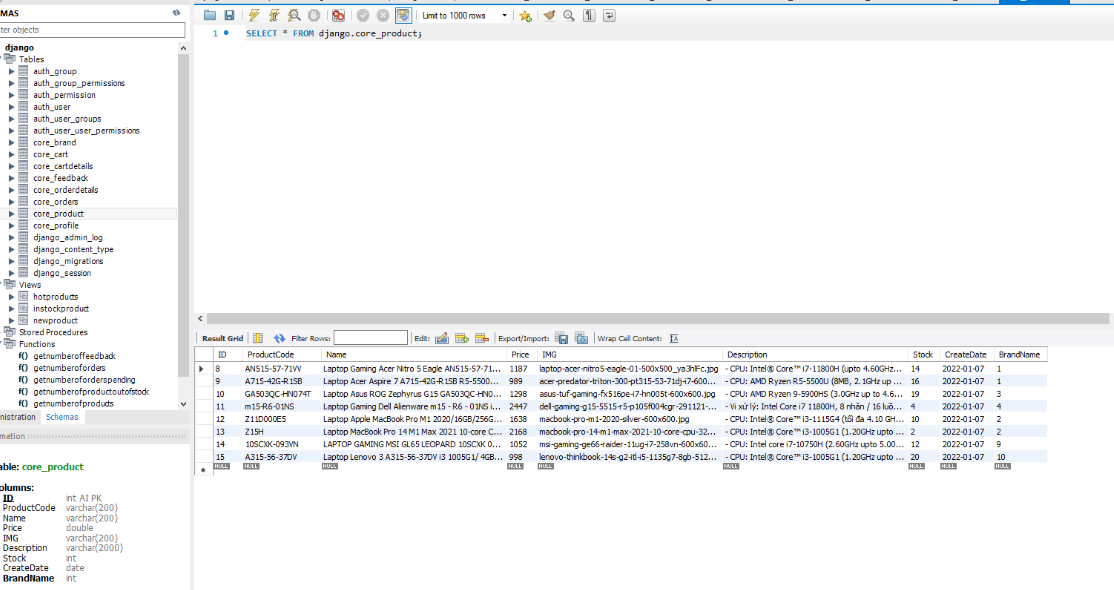


Màn hình Mysql sau khi được cài thành công !

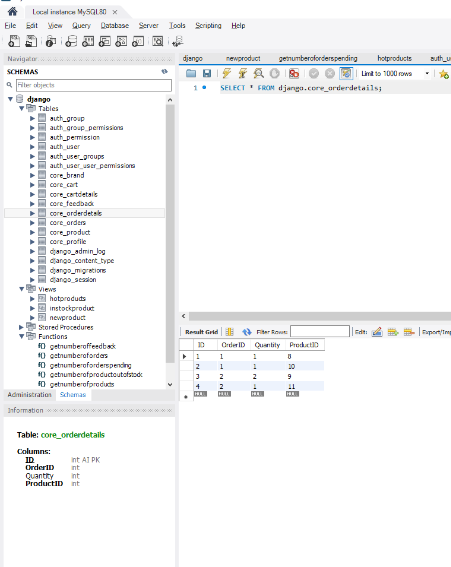
1. **Các bảng trong cơ sở dữ liệu**

*Có 7 bảng trong cơ sở dữ liệu*

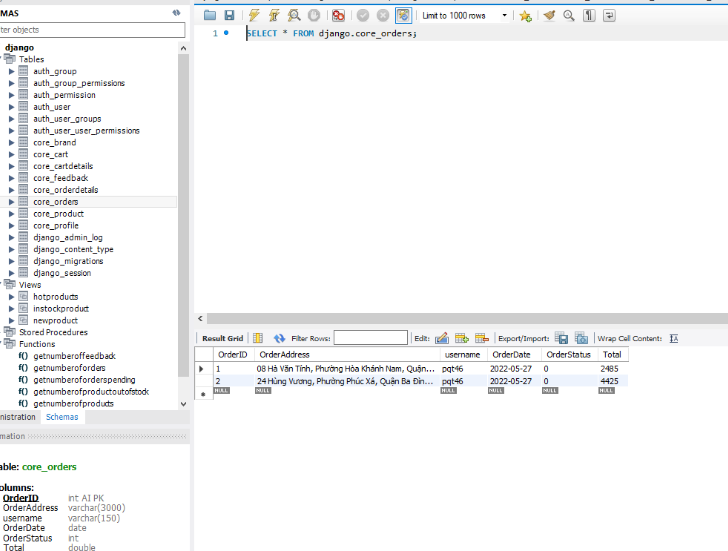
Đầu tiên là bảng core\_product: mô tả chi tiết các sản phẩm



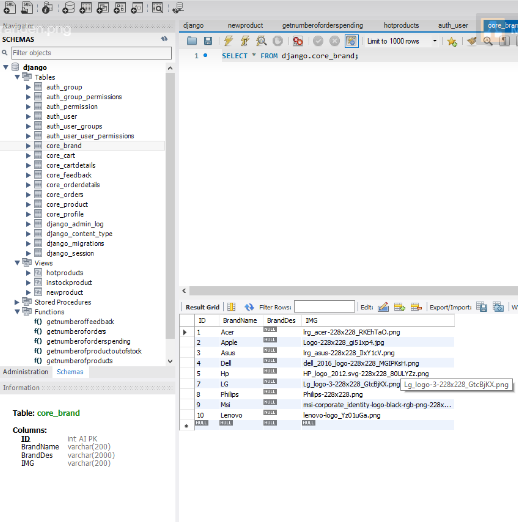
Bảng core\_orderdetails: hiển thị chi tiết đơn hàng mà khách hàng đã đặt



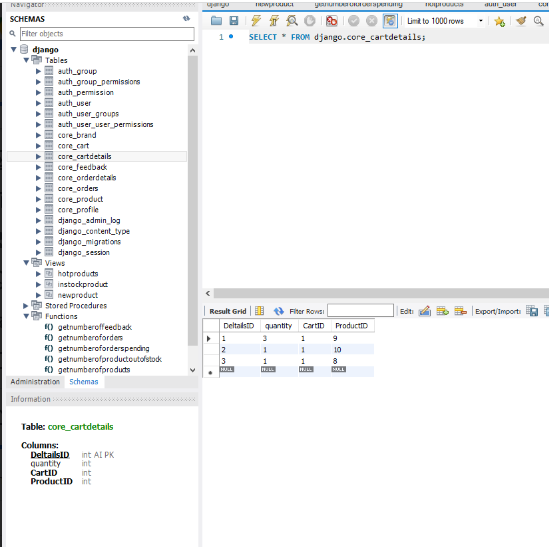
Bảng core\_orders: hiển thị các đơn hàng



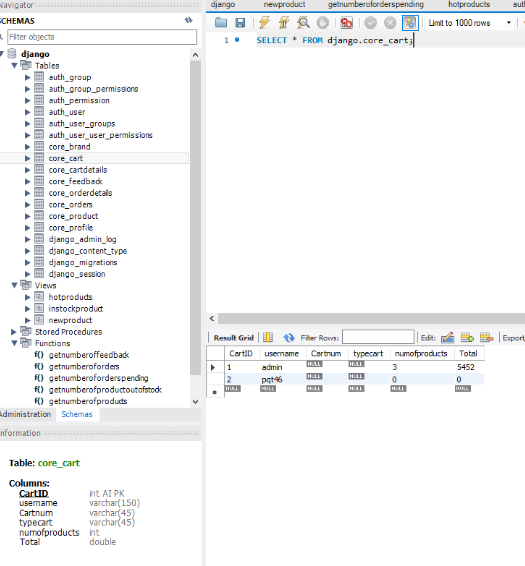
Bảng core\_brand: hiển thị các hãng của sản phẩm



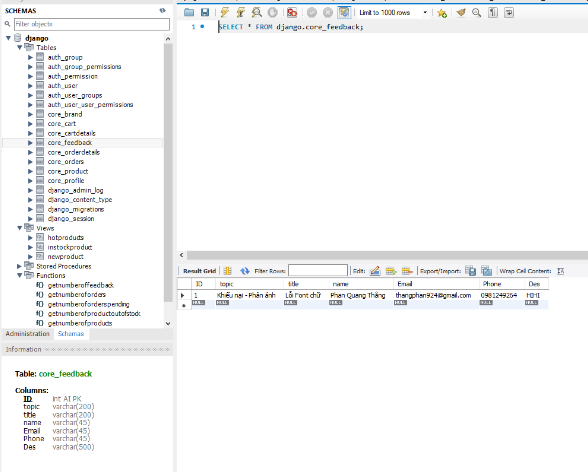
Bảng core\_cartdetails: hiển thị chi tiết giỏ hàng



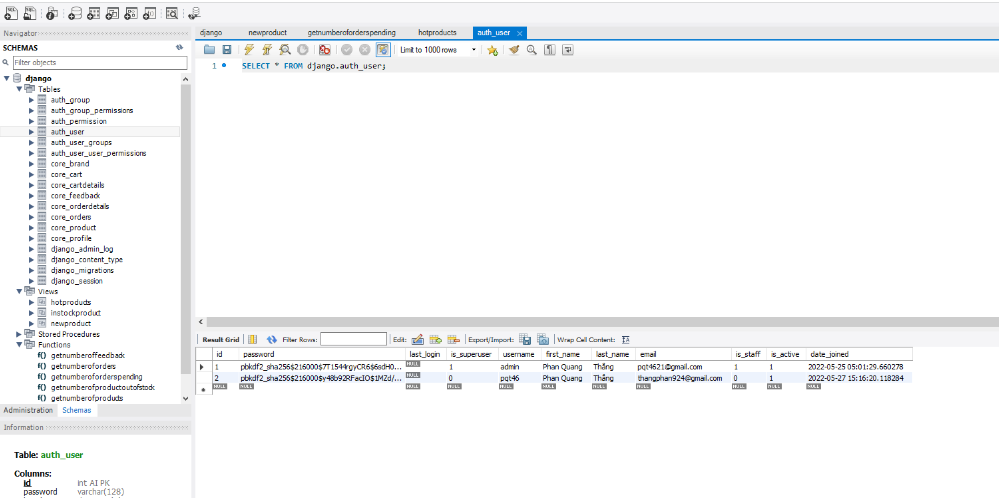
Bảng core\_cart: hiển thị giỏ hàng của khách hàng



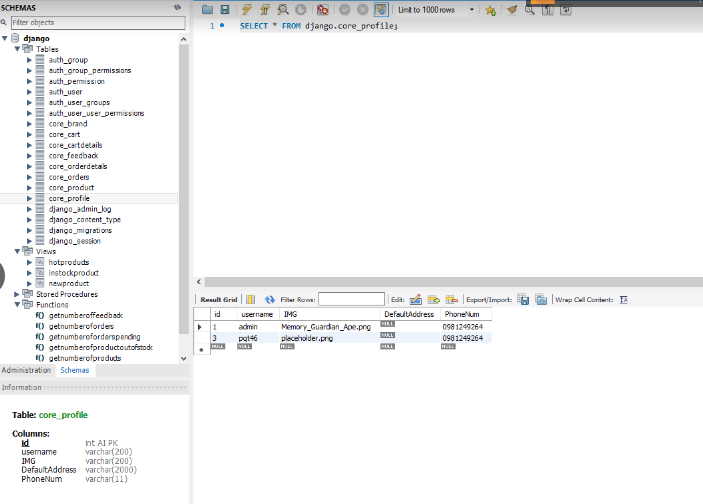
Bảng core\_feedback: hiển thị các đánh giá



Bảng auth\_user

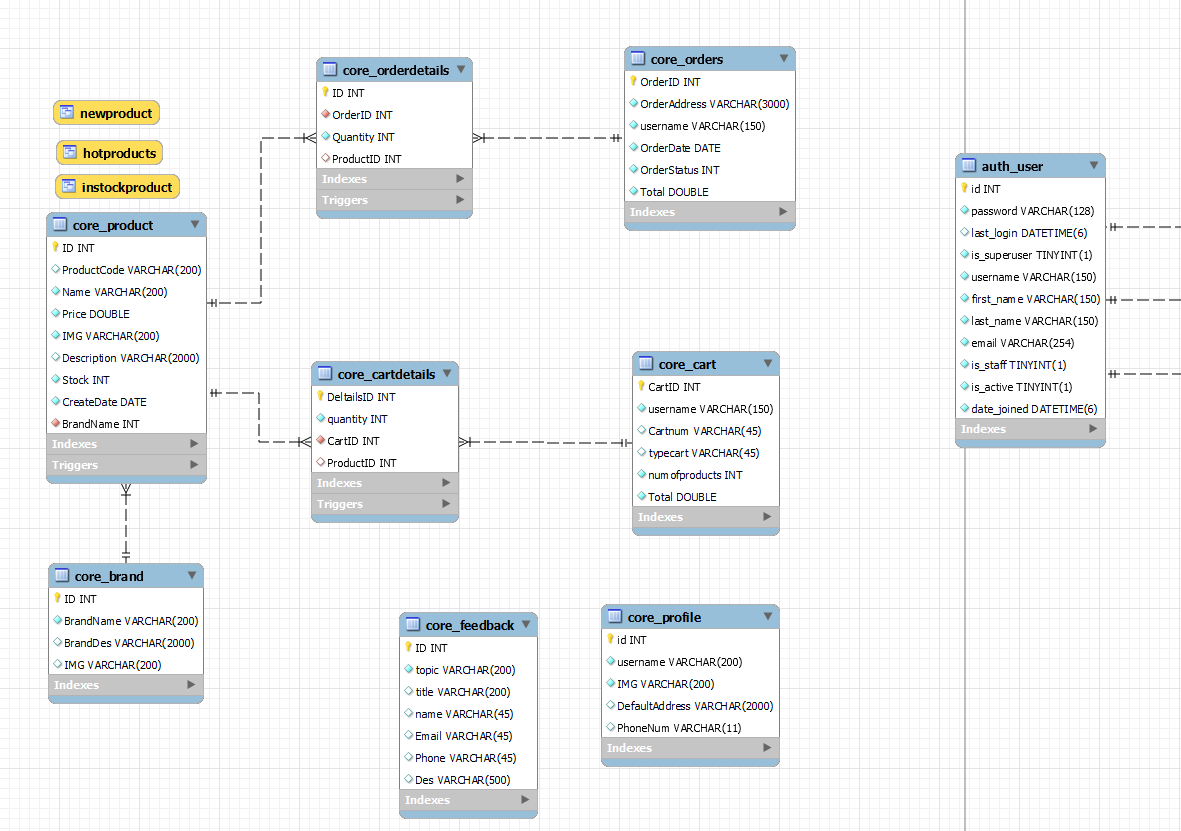


Bảng core\_profile

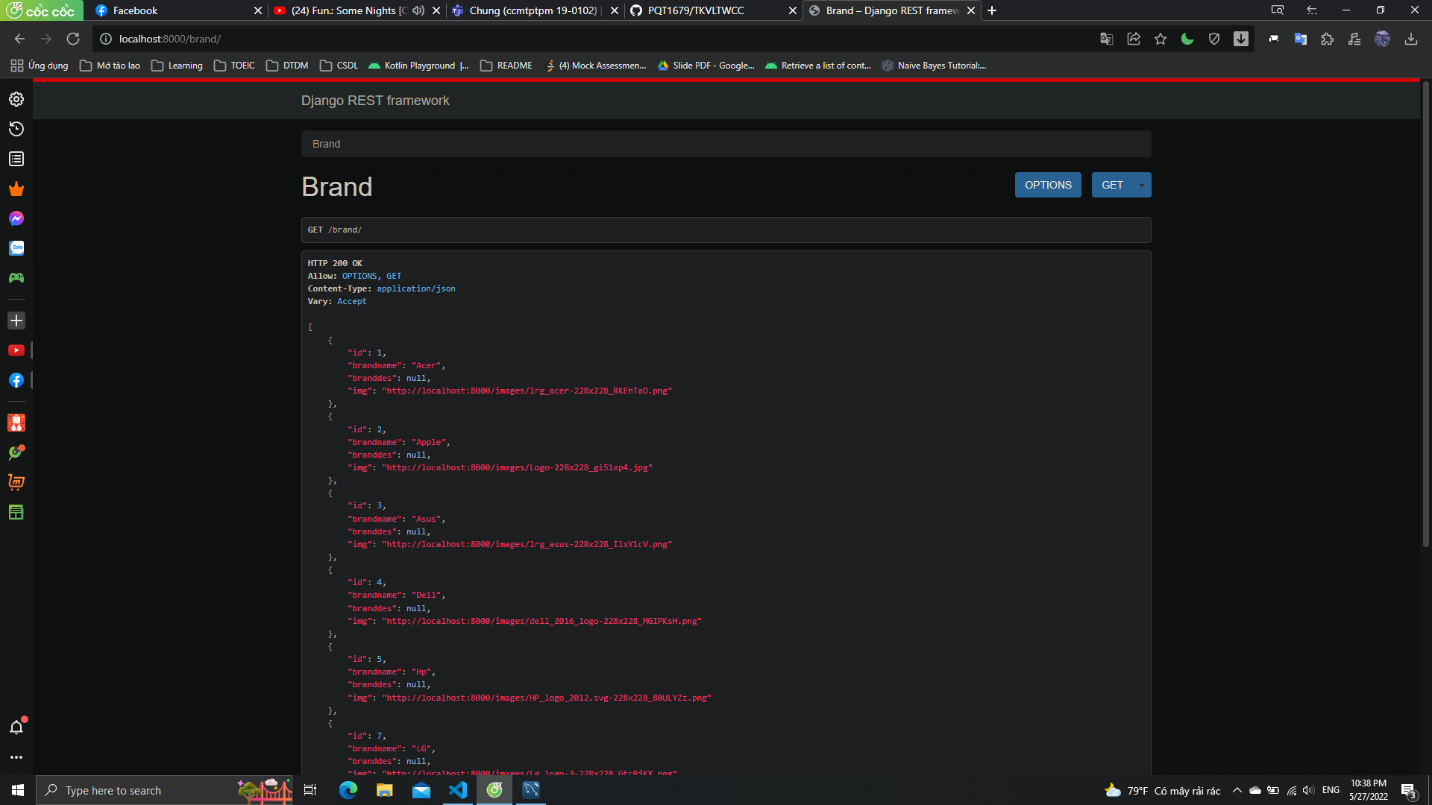


1. **Mối quan hệ giữa các bảng**

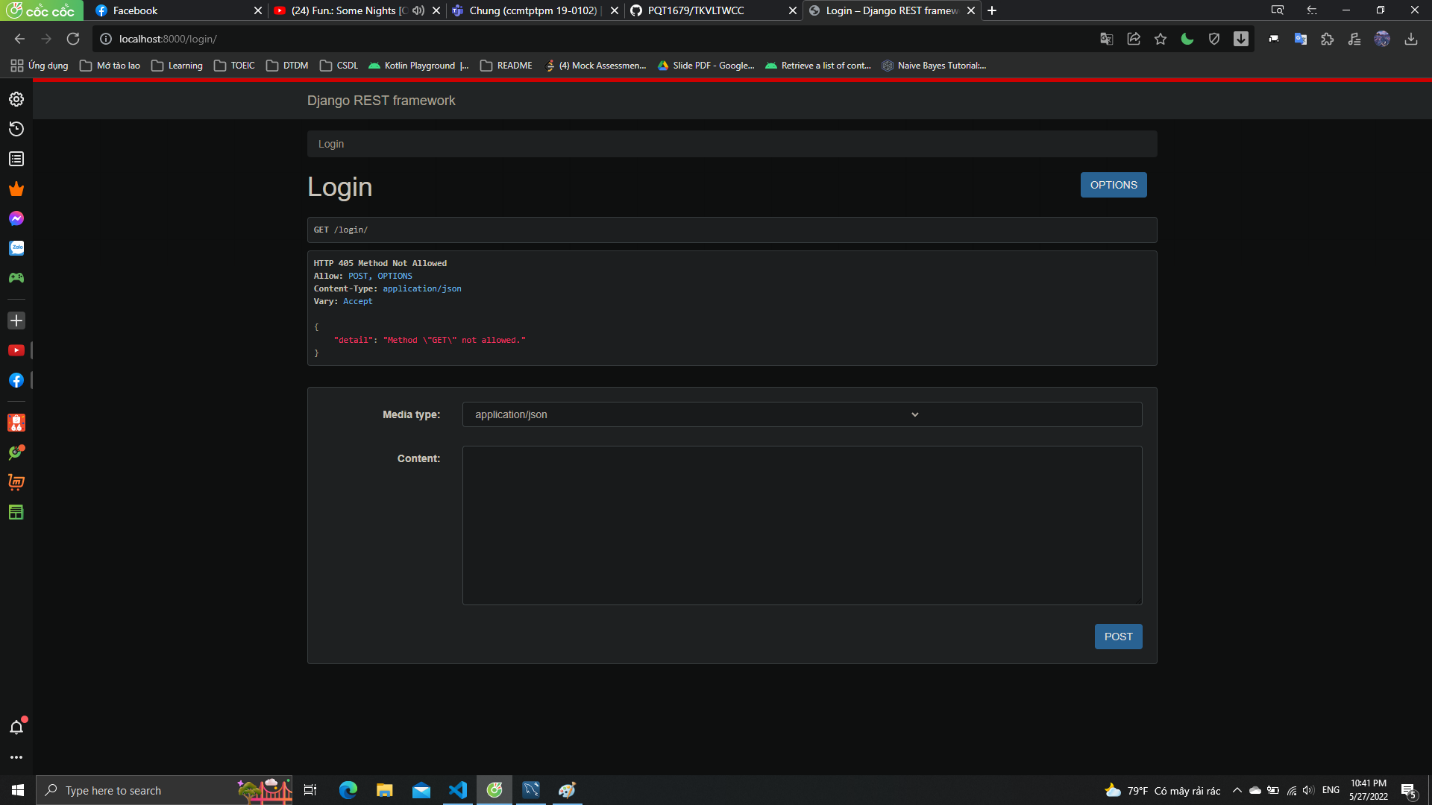
Mối quan hệ giữa các bảng được mô tả như sau



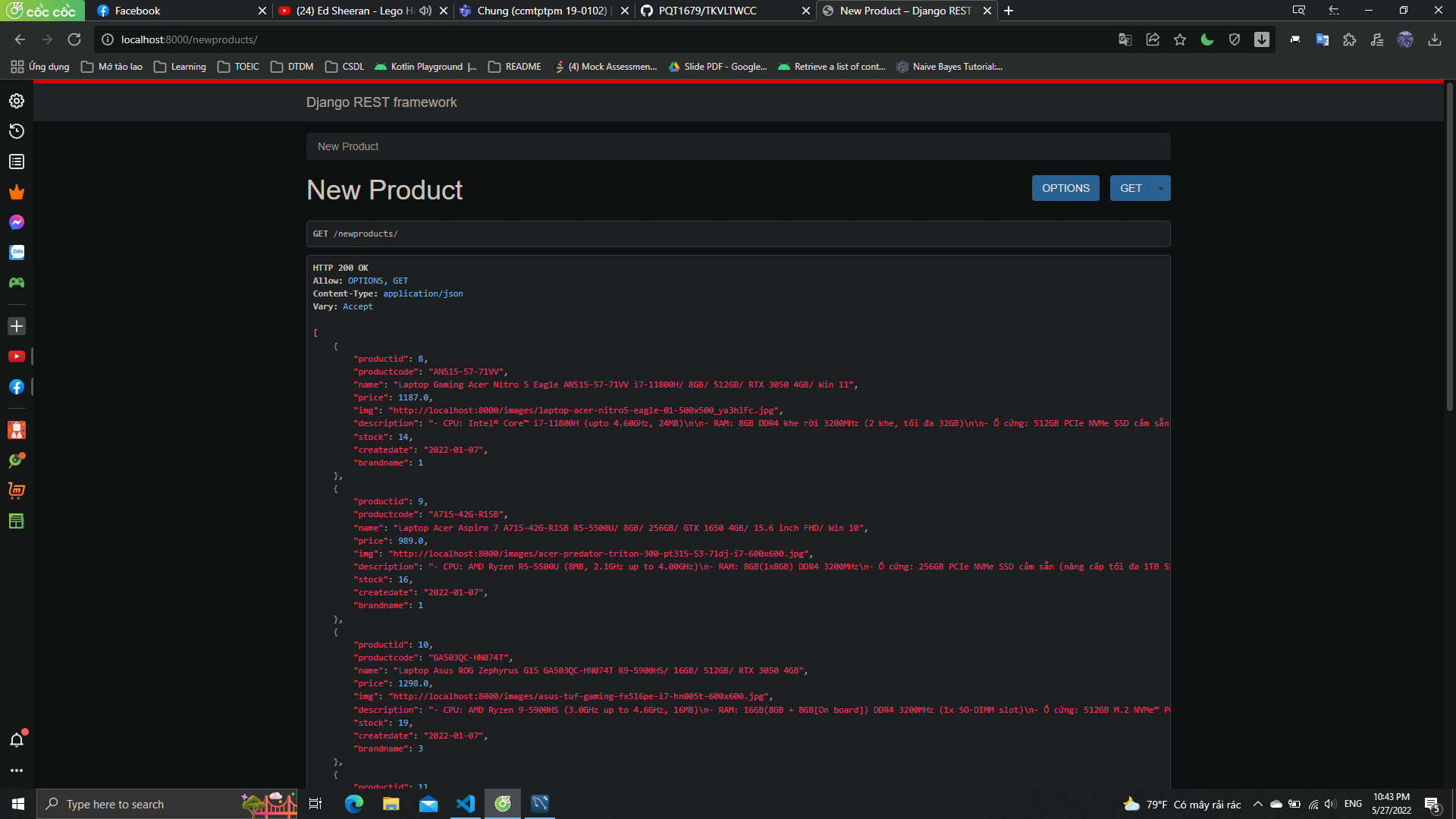
1. **Dữ liệu api của cơ sở dữ liệu tập trung**

****

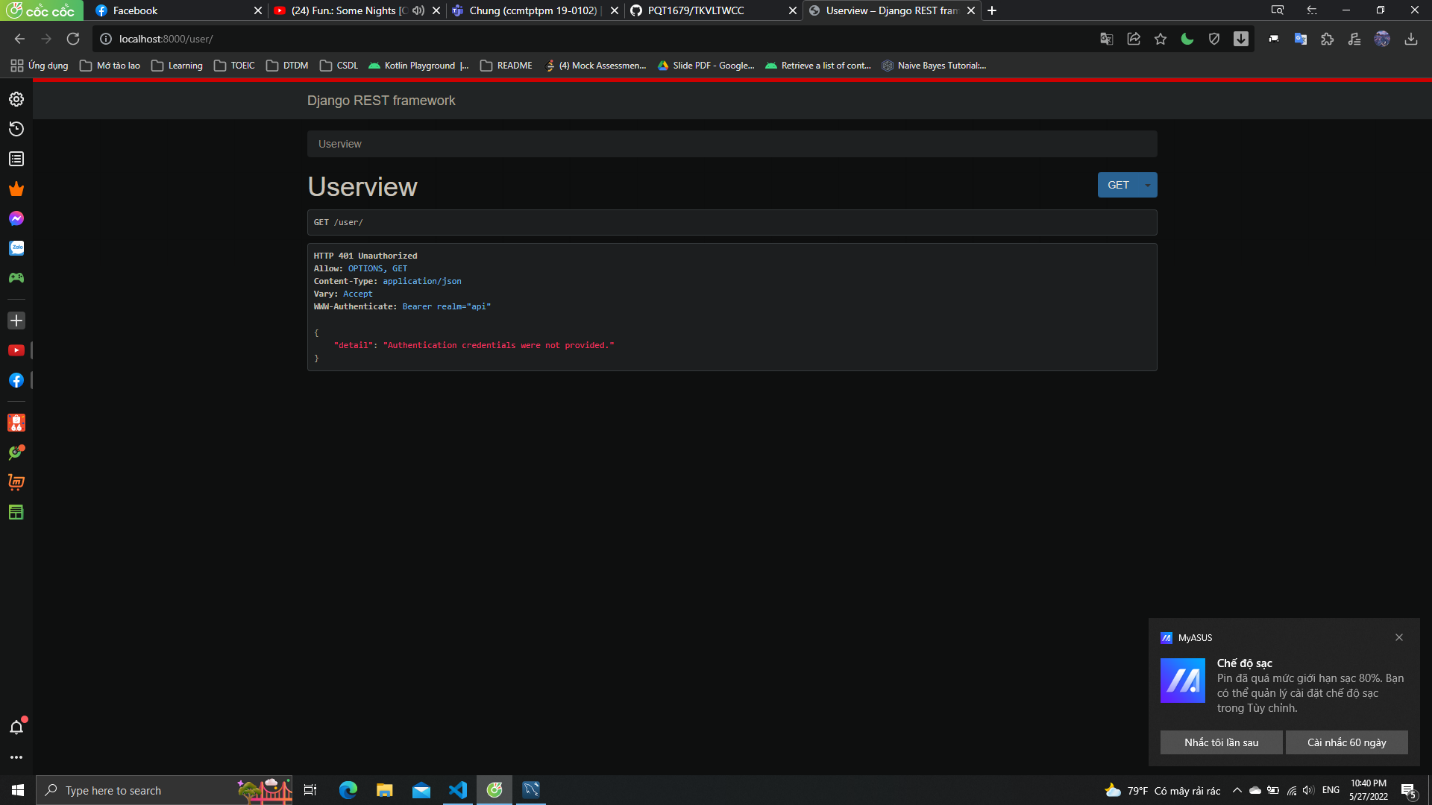
Dữ liệu API của Brand



Dữ liệu API của login



Dữ liệu API của new product

 Dữ liệu API của user

# CHƯƠNG 2: NGÔN NGỮ HỔ TRỢ

1. **Giới thiệu ngôn ngữ**
2. **Front-end**

Sử dụng các framwork

* Nuxtjs framework
* Vue
* Axios
* Css,boostrap
* Nuxtjs framework

**Khái niệm**: Đây là một framework của Vuejs được sử dụng để xây dựng những ứng dụng từ Vuejs. Nó có thể dùng để xây dựng một static landing pages hay thậm chí là những ứng dựng phức tạp.

**Một số tính năng nổi bật:**

* Viết dưới dạng file vue (.vue)
* Tự động split code để load javascript nhanh hơn
* Được áp dụng server side rendering
* Hỗ trợ routing khá dễ dàng
* Hỗ trợ ES6/ES7
* Đóng gói và nén javascript, css
* Quản lý các thẻ ở phần head
* Vue

**Khái niệm:** Vue.js là một framework linh động (nguyên bản tiếng Anh: progressive – tiệm tiến) dùng để xây dựng giao diện người dùng (user interfaces)

**Lý do nên sử dụng vue:** Nếu sử dụng Vue thì bạn sẽ không cần phải trải qua quá nhiều bước để build mà có thể đi thẳng vào vấn đề một cách dễ dàng. Bởi vì, Vue không cần sử dụng build tool quá phức tạp mới có thể xây dựng ứng dụng, bạn chỉ cần khai báo một script là có thể phát triển một ứng dụng bằng Vue.

* Vue có thể tạo cấu trúc project nhanh chóng hơn
* Hiện nay, tài liệu về Vue ngày càng đa dạng, rõ ràng về ngôn ngữ
* Axios

**Khái niệm:** một thư viện HTTP Client dựa trên Promise dành cho node.js và trình duyệt. Nó có tính đẳng hình (tức là cùng codebase có thể chạy trong cả trình duyệt và node.js). Ở phía server thì nó sử dụng native module http trong node.js, còn ở phía client (trình duyệt) thì nó sử dụng XMLHttpRequest.

**Một số tính năng:**

* Tạo request từ trình duyệt bằng XMLHttpRequest
* Tạo request từ node.js bằng http
* Hỗ trợ Promise API
* Đón chặn request và response
* Biến đổi dữ liệu request và response
* Bãi bỏ request
* Tự động chuyển đổi cho dữ liệu JSON
* Hỗ trợ phía client bảo vệ chống lại XSRF
* Css, boostrap: là một framework miễn phí giúp cho quá trình phát triển cũng như chỉnh sửa web được nhanh và dễ dàng hơn.

**Vì sao nên sử dụng boostrap**

Ưu điểm:

* Dễ sử dụng: Bất kỳ ai có kiến thức cơ bản về HTML và CSS đều có thể bắt đầu sử dụng Bootstrap.
* Các tính năng đáp ứng (Responsive features): responsive CSS của Bootstrap điều chỉnh cho điện thoại, máy tính bảng và máy tính để bàn.
* Cách tiếp cận Mobile-first: Trong Bootstrap 3, mobile-first styles là một phần của core framework.
* Khả năng tương thích trình duyệt: Bootstrap tương thích với tất cả các trình duyệt hiện đại (Chrome, Firefox, Internet Explorer, Edge, Safari và Opera).

Mục đích sử dụng các framework

* Tạo nên khung sườn chính cho cả trang web. Đặc biệt CSS, Bookstrap tạo nên sự sắp xếp cố định cùng những hiệu ứng bắt mắt.
* Tiến hành tiền xử lý trước khi gửi về phía server và hiển thị những thông báo thân thiện.
* Phân trang, hiển thị sản phẩm.

1. **Back-end**

Các framework sử dụng

* Django
* Django rest-framework
* Mysql
* Django

**Khái niệm**: là một trong số những web framework bậc cao miễn phí, là mã nguồn mở được tạo ra bởi ngôn ngữ Python dựa trên mô hình mô hình MTV (gồm Model-Template-Views).

**Ưu điểm**:

* Bảo mật tốt
* Mở rộng thỏa thích: Django có sẵn tính năng mở rộng nhằm hỗ trợ các lập trình viên quản lý lưu lượng người truy cập
* Dễ sử dụng: Django được tạo ra bởi ngôn ngữ lập trình Python và mô hình MVC nên rất dễ ứng dụng trong các dự án
* Dễ học: Có nhiều tài liệu hỗ trợ việc học Django, bao gồm cả tài liệu miễn phí trên mạng và sách in

**Nhược điểm:**

* Định tuyến tương đối khó
* Không đưa ra các cảnh báo khi xuất hiện lỗi trong mẫu.

**Vì sao nên sử dụng Django:**

* Nhanh chóng, đơn giản trong lập trình web/ứng dụng
* Đa dạng về tài liệu
* Sử dụng nguyên tắc DRY - không trùng lặp
* Hỗ trợ ORM, mang lại sự tương thích, khả năng hoạt động với các CSDL thường gặp như Oracle và SQL
* Linh hoạt cao: Xây dựng nhiều dạng website, ứng dụng cần thiết cho mọi lĩnh vực, tổ chức, cơ quan.
* Django rest-framework một trong những công cụ hổ trợ đắc lực trong việc xây dựng webAPI.
* Mysql

**Khái niệm:** MySQL là một hệ thống quản trị cơ sở dữ liệu mã nguồn mở (gọi tắt là RDBMS) hoạt động theo mô hình client-server. Được sử dụng rất phổ biến hiện nay

**Lợi ích của Mysql:**

* Là cơ sở dữ liệu tốc độ cao, ổn định
* Độ bảo mật thông tin cao
* Dễ sử dụng và có tính khả chuyển: tuy có tính năng cao nhưng thực sự, MySQL là một hệ thống cơ sở dữ liệu rất đơn giản và ít phức tạp .
* Có thể hoạt động trên nhiều hệ điều hành với lượng lớn các hàm tiện ích
* Hoàn toàn miễn phí

Mục đích sử dụng các framework

* Xử lý các thông tin từ các form hay các thông tin gửi từ Client.
* Lưu thông tin, phân quyền người dùng
* Lưu trữ thông tin.
* Tạo Api, truy cập cơ sở dự liệu

1. **Cách viết API**
2. **Khái niệm API**

API (Application Programming Interface) là một tập các quy tắc và cơ chế mà theo đó, một ứng dụng hay một thành phần sẽ tương tác với một ứng dụng hay thành phần khác. API có thể trả về dữ liệu mà bạn cần cho ứng dụng của mình ở những kiểu dữ liệu phổ biến như JSON hay XML.

Restful (REpresentational State Transfer) là một dạng chuyển đổi cấu trúc dữ liệu, một kiểu kiến trúc để viết API. Nó sử dụng phương thức HTTP đơn giản để tạo cho giao tiếp giữa các máy. Vì vậy, thay vì sử dụng một URL cho việc xử lý một số thông tin người dùng, REST gửi một yêu cầu HTTP như GET, POST, DELETE, vv đến một URL để xử lý dữ liệu.

RestAPI là một tiêu chuẩn dùng trong việc thiết kế các API cho các ứng dụng web để quản lý các resource. RESTful là một trong những kiểu thiết kế API được sử dụng phổ biến ngày nay để cho các ứng dụng (web, mobile…) khác nhau giao tiếp với nhau.

1. **Cách viết API cho web sử dụng Diango Rest Framework**

Có thể tham khảo tại link sau:

<https://www.digitalocean.com/community/tutorials/how-to-build-a-universal-application-with-nuxt-js-and-django>

**CHƯƠNG 3: PHÂN TÍCH THIẾT KẾ HỆ THỐNG WEBSITE**

1. **Giới thiệu về website**

Ngày nay với sự phát triển cũng như Internet đã trở thành dịch vụ phổ biến và thiết yếu có ảnh hưởng sâu rộng đến thói quen sinh hoạt, giải trí của nhiều người Cùng với sự phát triển nhanh chóng của Internet thì các hình thức mua và bán hàng hóa cho mọi người ngày càng đa dạng và phát triển hơn. Các ứng dụng Web ngày càng trở nên phổ biến. Website sẽ dần dần thay thế những phương thức kinh doanh cũ trong các doanh nghiệp bởi tính ưu việc mà website mang lại như: nhanh hơn, tiện lợi hơn, hiệu quả hơn và không bị giới hạn về không gian và thời gian đặc biệt là trong mua dịch này.

Trước nhu cầu đó, nhóm quyết định chọn và thực hiên đề tài Xây dựng Website bán hàng trực tuyến, cụ thể là website bán laptop dựa trên cơ sở dữ liệu tập trung

1. **Phân tích thiết kế hệ thống**
2. **Phân tích chức năng**

* *Về phía khách hàng*

Các chức năng chính:

* Đăng nhập
* Đăng ký
* Đăng xuất
* Chỉnh sửa thông tin khách hàng
* Xem danh mục hãng, danh mục sản phẩm
* Xem chi tiết thông tin sản phẩm
* Tìm kiếm thông tin
* Thêm sản phẩm vào giỏ hàng
* Chi tiết giỏ hàng
* Thanh toán
* Gửi Feedback
* *Về phía Admin*

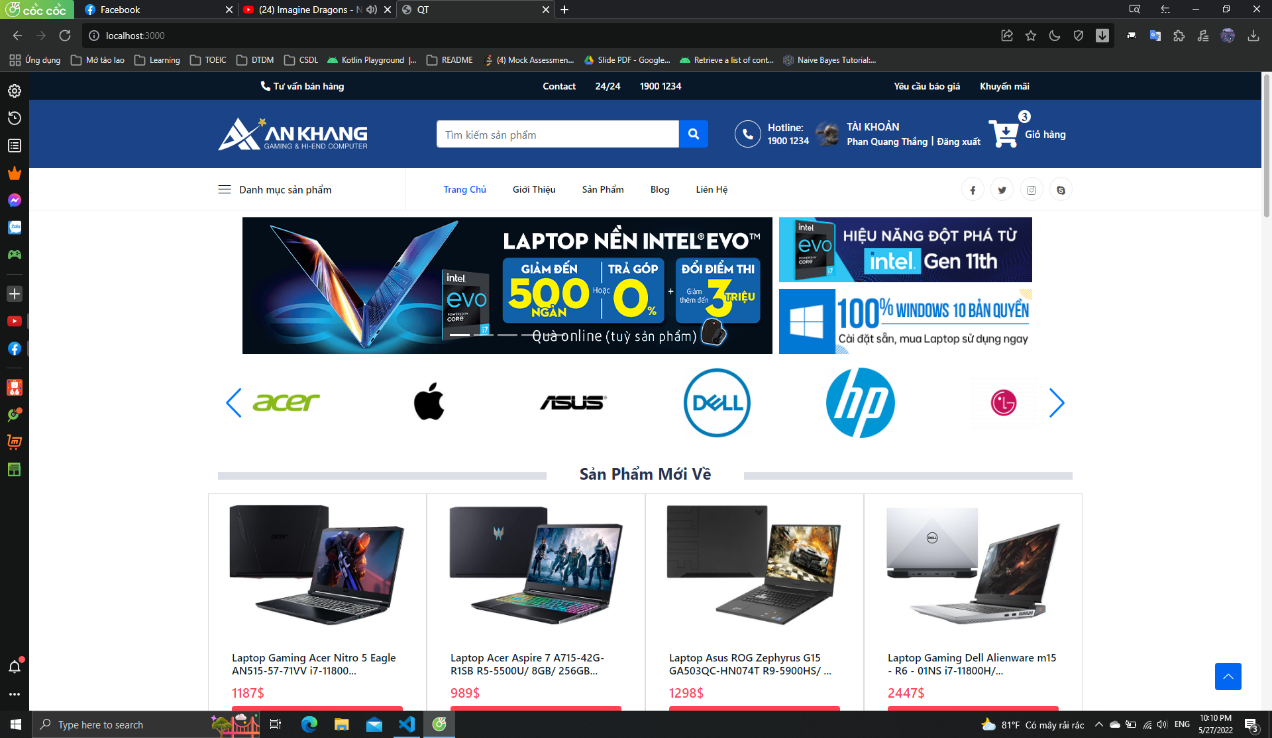
Các chức năng chính:

* Đăng nhập
* Đăng xuất
* Quản lý thành viên
* Quản lý sản phẩm: Thêm, sửa, xóa sản phẩm
* Quản lý user
* Quản lý đơn hàng
* Quản lý phản hồi

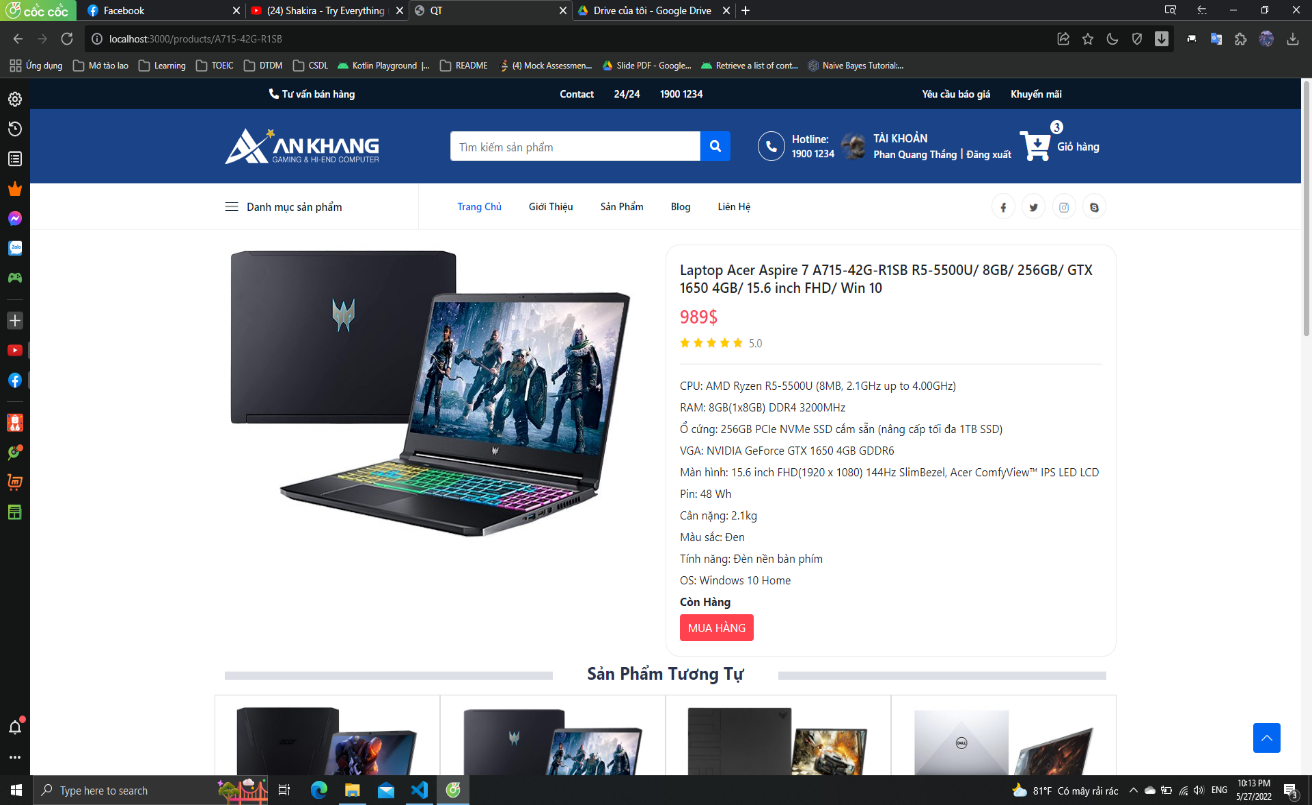
1. **Các yêu cầu về hệ thống**

* Giao diện hệ thống phải dễ sử dụng, trực quan, thân thiện với người dùng
* Hệ thống phải xử lý nhanh chóng và chính xác
* Tính bảo mật và độ an toàn phải được đảm bảo
* Tương thích với đa phần các trình duyệt web hiện tại

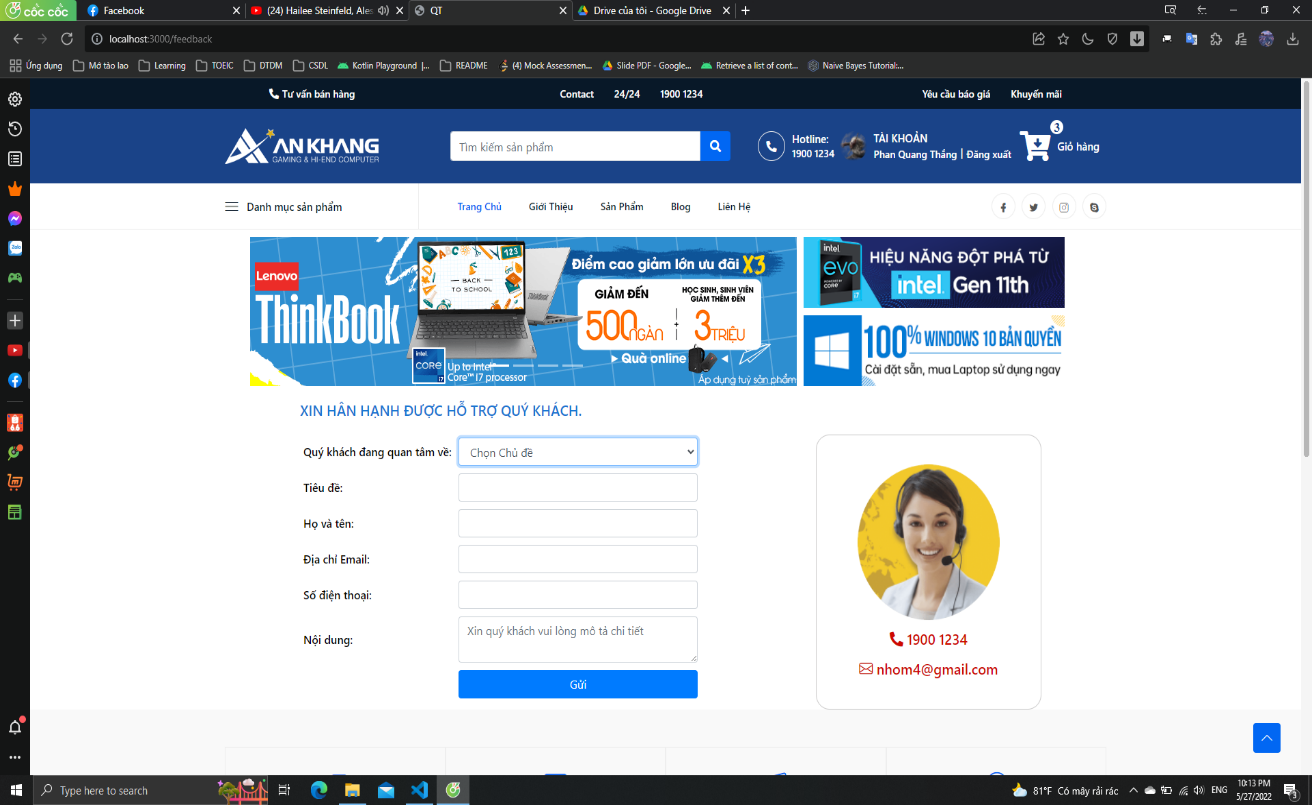
1. **Sản phẩm hoàn thiện**



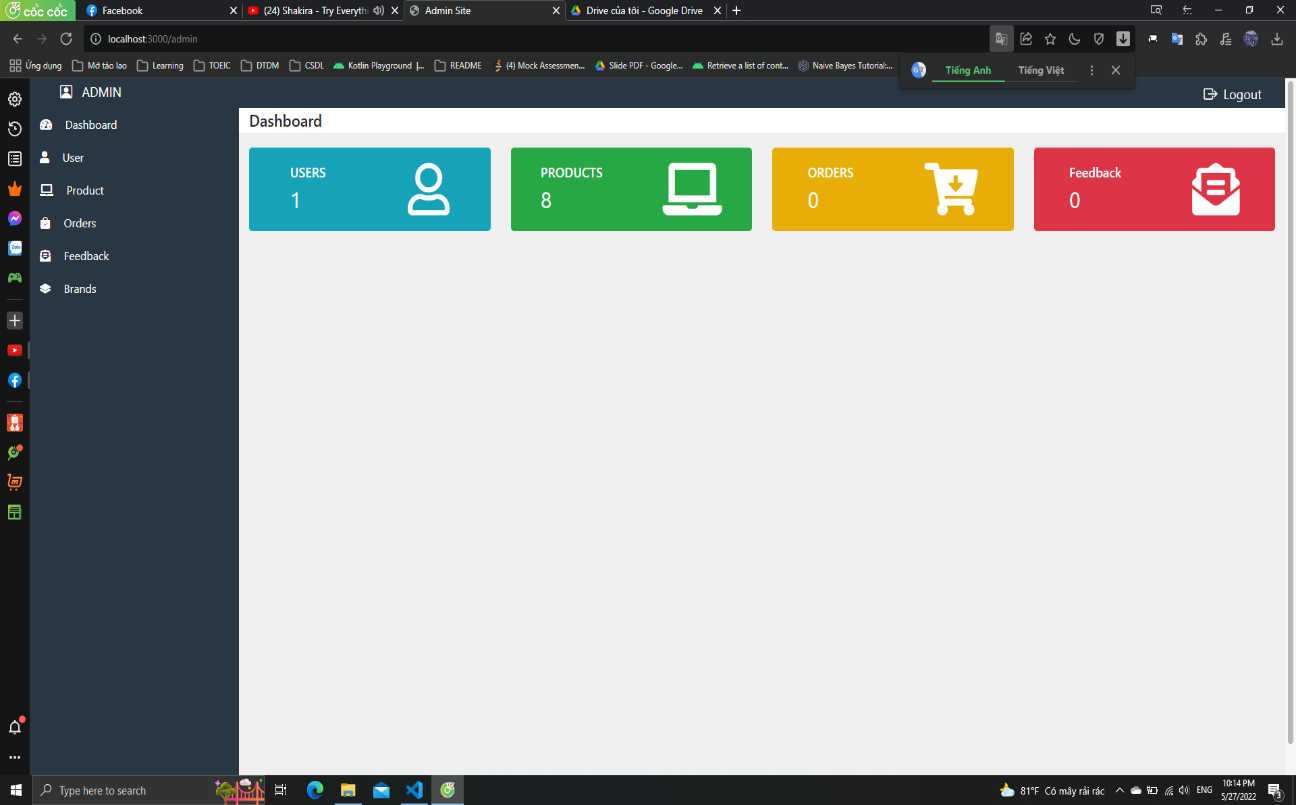
*Hình 1: Trang chủ của website*



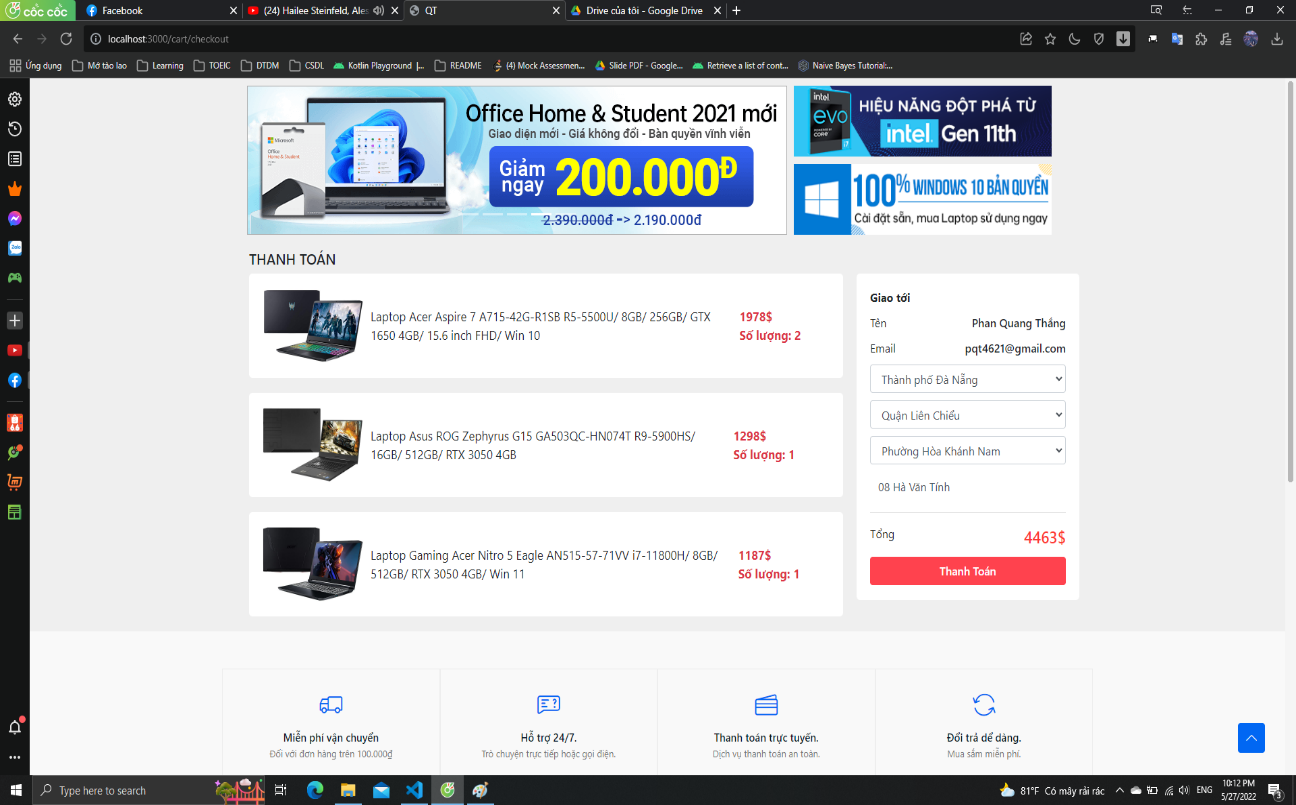
*Hình 2: trang chi tiết sản phẩm*



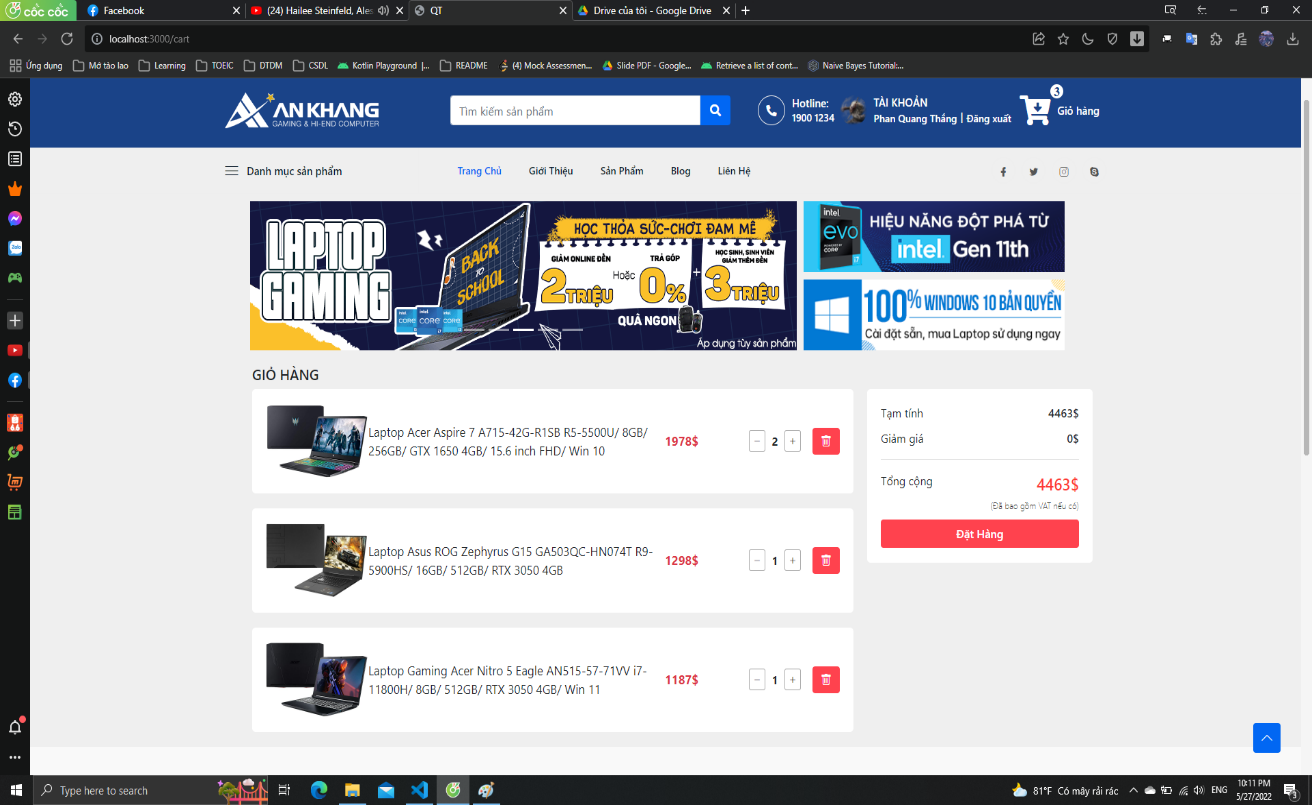
*Hình 3: Trang Feedback*



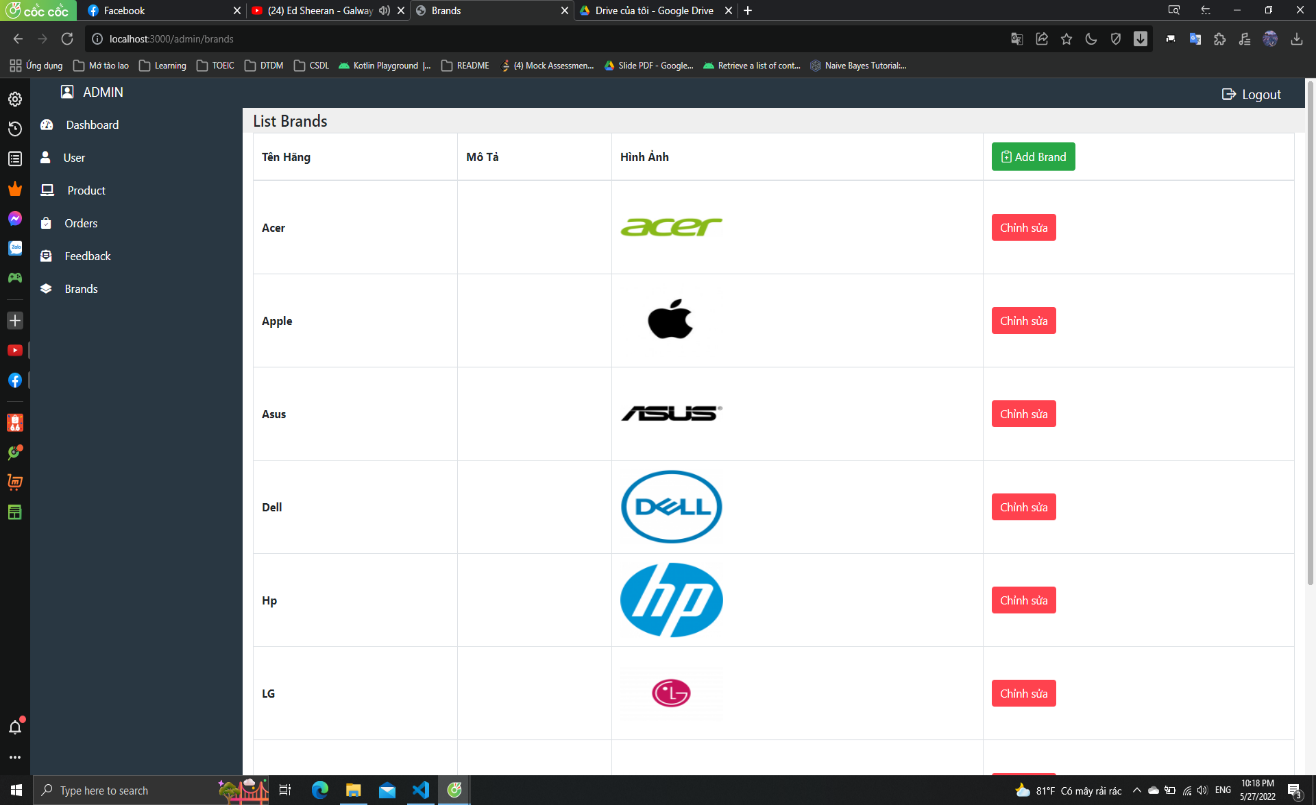
*Hình 4: Giao diện Admin*



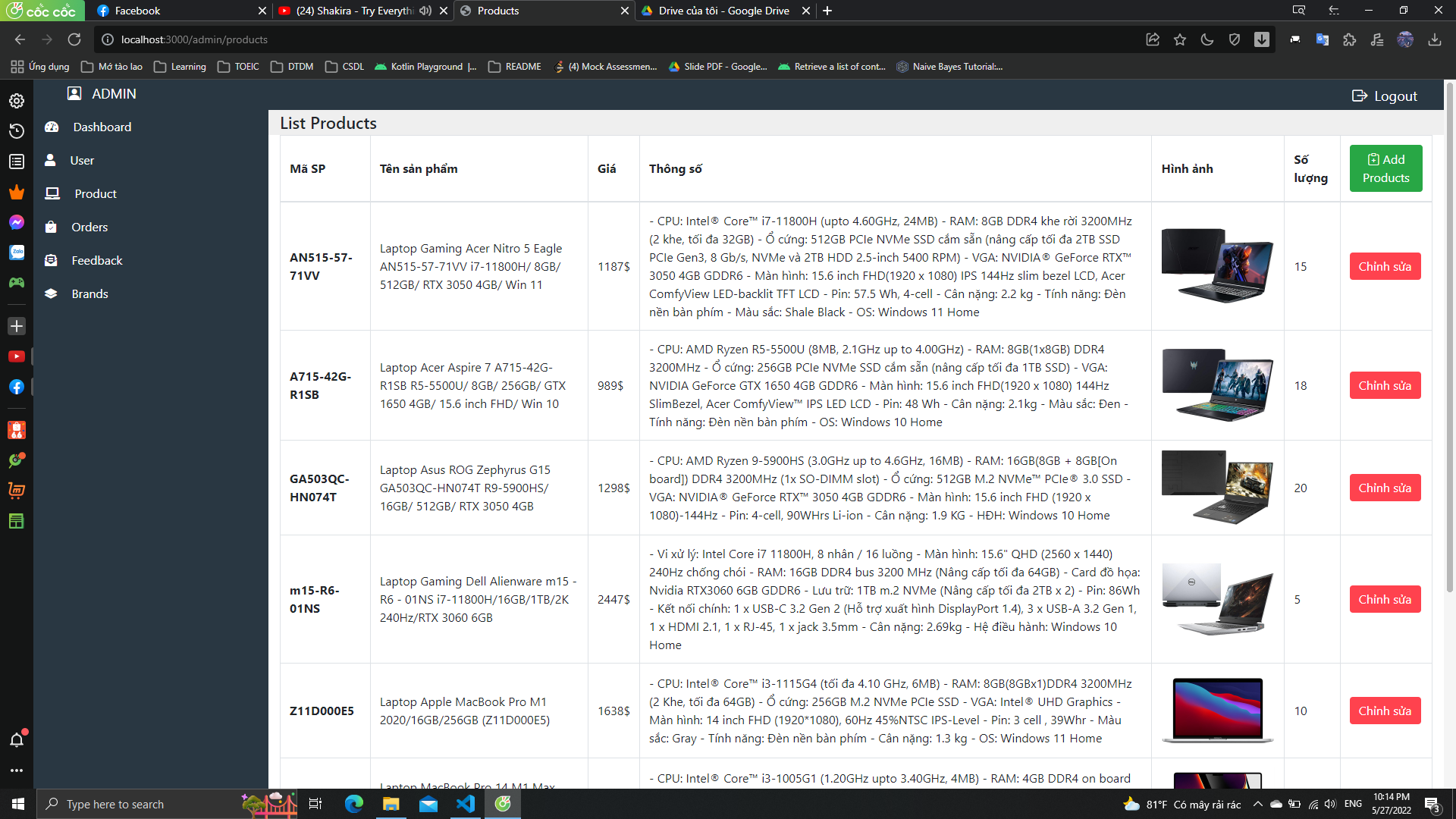
*Hình 5: Giao diện checkout*



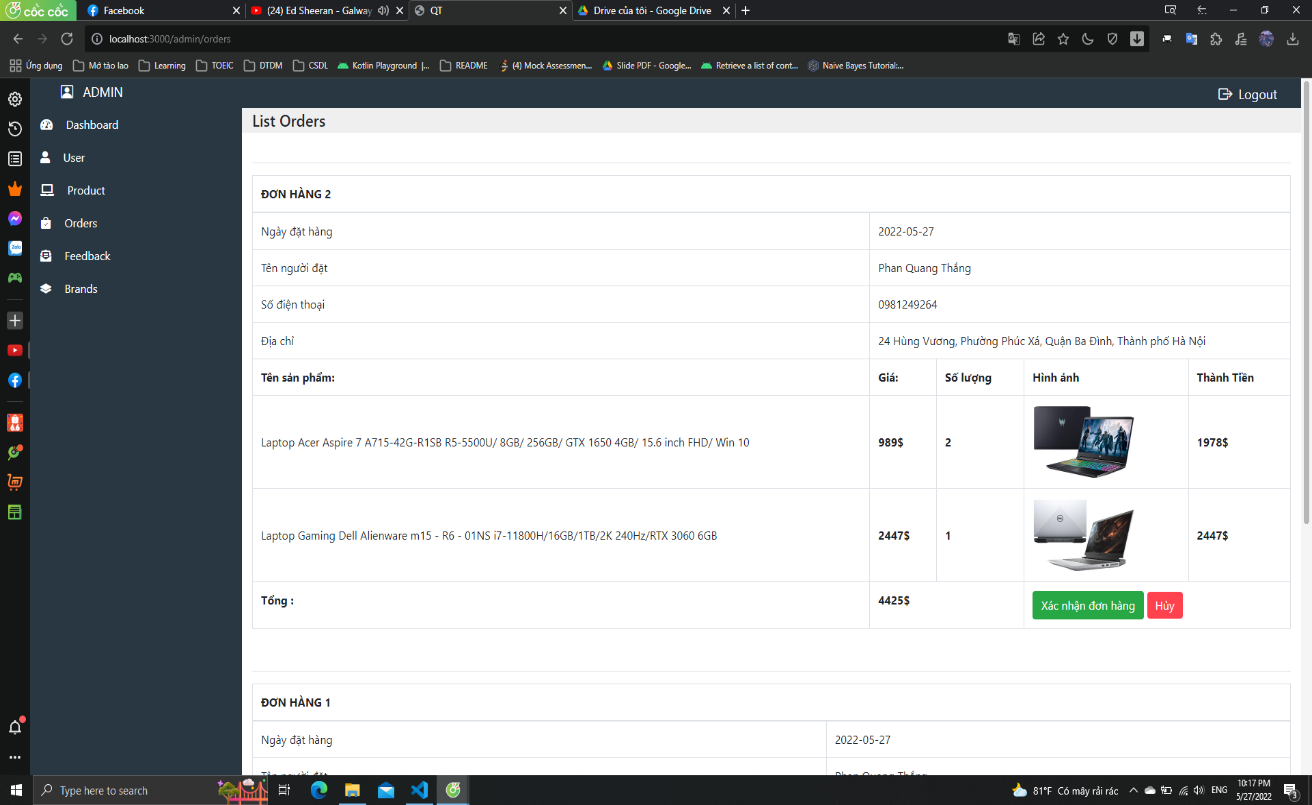
*Hình 6: giao diện giỏ hàng*



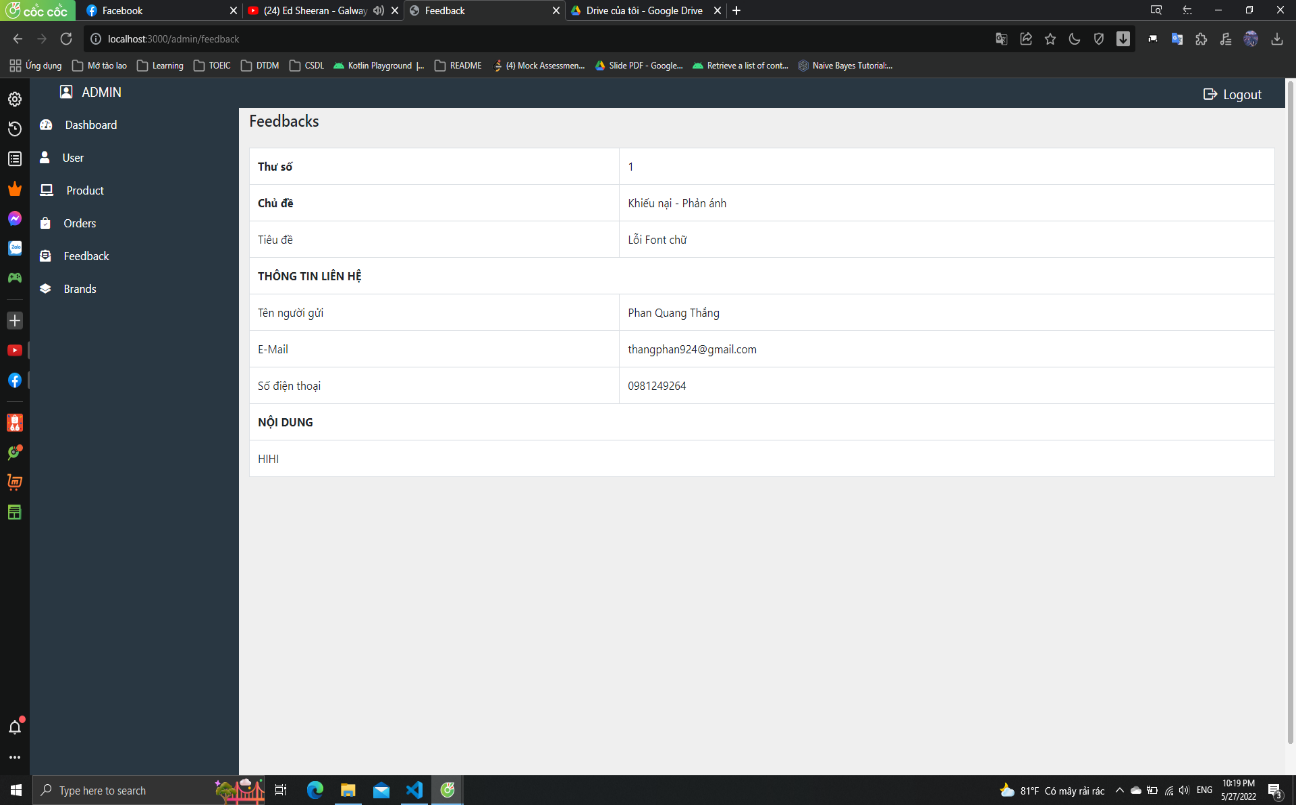
*Hình 7: giao diện các hãng*



*Hình 8: giao diện sản phẩm cho admin*



*Hình 9: giao diện order cho admin*



*Hình 10: giao diện feedback cho admin*

**KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN**

Trong đồ án này chúng em chỉ tiếp cận cơ sở dữ liệu tập trung ở mức đơn giản nên sau khi hoàn thành đồ án môn học CSDL-NC này chúng em sẽ cố gắng tìm tòi tiếp cận cơ sở dữ liệu tập trung cao hơn nhờ sự giúp đỡ của các thầy cô ở trường cũng như nguồn tài liệu khác để có thể nâng cao tốt hơn nhằm phục vụ cho quá trình học tập và công việc.

Hướng phát triển tiếp theo của chúng em đó là xây dựng thêm các chức năng còn thiếu phát triển cơ sở dữ liệu, tối ưu hóa cơ sở dữ liệu

Cơ sở dữ liệu tập trung là cơ sở dữ liệu dựa trên web này lưu trữ dữ liệu ở một vị trí trung tâm. Qua đó giúp cấu hình cho phép nhiều người dùng dễ dàng truy cập thông tin từ xa. Đồng thời cơ sở dữ liệu tập trung dễ dàng cấu hình và quản lý hơn.

Trong quá trình thực hiện đồ án. Nhóm em xin chân thành cảm ơn giảng viên Võ Công Đình đã tận tình hướng dẫn và giúp đỡ hoàn thiện đồ án này.

# TÀI LIỆU THAM KHẢO

[1] <https://www.django-rest-framework.org/>

[2] <https://www.python.org/>

[3] <https://nodejs.org/en/docs/>

[4] <https://docs.djangoproject.com/en/4.0/>

[5]<https://www.digitalocean.com/community/tutorials/how-to-build-a-universal-application-with-nuxt-js-and-django>

[6] <https://nuxtjs.org/>