**ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH**

**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN**

🙤🙧🟍🙥🙦



**WED THƯƠNG MẠI ĐIỆN TỬ**

**Kỹ thuật lập trình python – IE221.M22.CNCL**

|  |  |  |
| --- | --- | --- |
| Sinh viên thực hiện: | | |
| STT | Họ tên | MSSV |
| 1 | Nguyên Huy Phong | 19522011 |
| 2 | Hồ Sĩ Luân | 19521807 |

**TP. HỒ CHÍ MINH – 5/2022**

**Mục Lục**

[1. GIỚI THIỆU 0](#_Toc103769620)

[2. NỘI DUNG 1](#_Toc103769621)

[2.1. Sử dụng các mẫu font-end thông dụng, bắt mắt để người dùng để tiếp cận. 1](#_Toc103769622)

[2.2. Cài đặt môi trường và các thư viện cần thiết. 1](#_Toc103769623)

[2.3. Tạo project django mới và cho các file html, css, js vào trong project. 1](#_Toc103769624)

[2.4. Tạo các đường dẫn cần thiết tới các trang html. 2](#_Toc103769625)

[2.5. Tạo database cho trang web. 4](#_Toc103769626)

[2.6. Tạo trang admin cho trang web 5](#_Toc103769627)

[2.7. Hiển thị dữ liệu ra ngoài trang web. 5](#_Toc103769628)

[2.8. Thêm bài viết và chức năng bình luận. 5](#_Toc103769629)

[2.9. Xóa các chức năng không cần thiết trên mẫu html sẵn và làm cho các nút trên mẫu hoạt động. 6](#_Toc103769630)

[2.10. Sử dụng vue js để thể hiện những thứ cần thời gian thực: ví dụ thêm xóa sửa trong giỏ hàng. 6](#_Toc103769631)

[2.11. Tạo người dùng và chức năng đăng nhập, đăng ký, đăng xuất. 6](#_Toc103769632)

[2.12. Dùng cookie để cho khách hàng chưa đăng nhập có thể mua hàng thêm giỏ hàng. 6](#_Toc103769633)

[2.13. Thêm hình thức thanh toán trực tuyến. 7](#_Toc103769634)

[2.14. Thêm chứ năng đánh giá sản phẩm. 7](#_Toc103769635)

[2.15. Thêm chức năng tìm kiếm sản phẩm. 7](#_Toc103769636)

[2.16. Thêm chức năng quản lý các đơn hàng đã được đặt. 7](#_Toc103769637)

[2.17. Deploy trang web lên internet. 8](#_Toc103769638)

[3. KẾT LUẬN 9](#_Toc103769639)

[PHỤ LỤC PHÂN CÔNG NHIỆM VỤ 11](#_Toc103769640)

# GIỚI THIỆU

Đề tài Web thương mại điện tử là đề tài xây dựng một hệ thống backend cho một trang web thương mại điện tử. Với yêu cầu là phải tạo ra hầu hết các chức năng mà một web thương mại có.

Thông qua đề tài này chúng em hướng đến một trang thương mại điện tử có thể sử dụng một cách dễ dàng, tiện lợi.

Cách làm là sử dụng các khóa học trên youtube,trên các website về django để học về django sau đó áp dụng kiến thức đó để tạo ra sản phẩm.

Các công cụ hỗ trợ gồm: Visual studio code, Github, Heroku.

Chúng em đã thành công xây dựng khá tốt một trang web như các mục tiêu đã đặt ra. Tuy nhiên còn nhiều chức năng chưa thực sự tối ưu. Hệ thống code cũng chưa quá khoa học.

Trong báo này, chúng em tập trung trình bày hai nội dung chính: (1) Quy trình thực hiện các bước đã đặt ra. (2) Đánh giá trang web và kết luận

Video demo: <https://www.youtube.com/watch?v=aDFr9RuXhVA>

Link trang web: <https://shopthuongmai.herokuapp.com/>

# NỘI DUNG

Diagram

Description automatically generated

Hình Sơ đồ được sử dụng khi thiết kế

## Sử dụng các mẫu font-end thông dụng, bắt mắt để người dùng để tiếp cận.

* Nhóm em sử dụng mẫu template ở đây: [www.templatebazaar.in/demo/fashion\_shop/](http://www.templatebazaar.in/demo/fashion_shop/)

## Cài đặt môi trường và các thư viện cần thiết.

* Cài đặt python
* Cài đặt pip
* Cài đặt virtualenv
* Cài đặt taggit

## Tạo project django mới và cho các file html, css, js vào trong project.

* Tạo thư mục static và thêm tất cả các file trong assets vào trong static.

Graphical user interface, application

Description automatically generated

Hình Thư mục static

* Tạo thư mục templates và cho tất cả file html vào trong.

A screenshot of a computer

Description automatically generated with medium confidence

Hình Thư mục templates

## Tạo các đường dẫn cần thiết tới các trang html.

* Tạo các hàm trong view có giá trị trả về các trang html trong file views.py

Graphical user interface, text, application, chat or text message

Description automatically generated

Hình File view.py

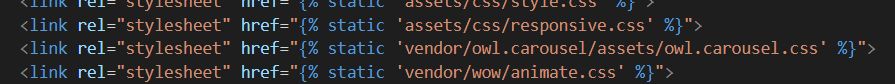
* Tạo các đường dẫn trỏ tới các hàm trong views.py trong file urls.py

Text

Description automatically generated

Hình File urls.py

* Trong thư mục templates tạo 1 file main.py để chứa header footer của trang web.
* Cho các file html khác extend từ file main.py.
* Chỉnh sửa các đường dẫn tới các file css, js, image



Hình 6 Chỉnh sửa đường dẫn tới file trong static

## Tạo database cho trang web.

* Tạo database trong file models.py tương tự như hình dưới

Diagram

Description automatically generated

Hình Sơ đồ database

## Tạo trang admin cho trang web

Text

Description automatically generated

Hình File admin.py

* Thêm các thực thể vào trong trang admin như hình trên

## Hiển thị dữ liệu ra ngoài trang web.

* Dùng các dòng lặp for để biểu diễn sản phẩm, bài viết, danh mục, bình luận, sản phẩm trong giỏ hàng. Các biến sẽ được chuyền thông qua các hàm bên trong views.py

## Thêm bài viết và chức năng bình luận.

* Chức năng gửi bình luận chỉ xuất hiện khi người dùng đẵ đăng nhập. Bình luận cho bài viết có 2 khóa ngoại là người dùng và bài viết. Còn bình luận cho sản phẩm có 2 khóa ngoại là người dùng và sản phẩm

## Xóa các chức năng không cần thiết trên mẫu html sẵn và làm cho các nút trên mẫu hoạt động.

* Thêm các đường dẫn vào các nút trong trang web

Text

Description automatically generated

Hình Thêm đường dẫn vô các nút đăng xuất, đăng nhập, đăng ký

## Sử dụng vue js để thể hiện những thứ cần thời gian thực: ví dụ thêm xóa sửa trong giỏ hàng.

* Thêm vào thẻ button hàm addToCart



Hình Thêm vào thẻ button @click với hàm addToCart

* Trong hàm add to cart thì ta sẽ gửi dữ liệu tới backend qua đường /update\_item/, với phương thức POST, dữ liệu truyền đi bao gồm csrftoken, id của sản phẩm và hành động muốn thực hiện. Còn ở phần frontend sẽ dùng js để thay đổi số lượng trên giỏ hàng.
* Thêm hàm tương tự cho nút giảm số lượng sản phẩm trong giỏ hàng
* Dữ liệu sẽ được truyền tới hàm updateItem trong view.py. Trong hàm updateItem sẽ sử lý yêu cầu và lưu dữ liệu

## Tạo người dùng và chức năng đăng nhập, đăng ký, đăng xuất.

* Sử dụng các form đăng nhập đăng ký có sẵn trong django. Nếu đăng ký thành công sẽ tự động đăng nhập người dùng và chuyển tới trang home.

## Dùng cookie để cho khách hàng chưa đăng nhập có thể mua hàng thêm giỏ hàng.

* Thêm hàm tạo cookie trên thẻ head trong file main.html
* Trong hàm addToCart thêm vào chức năng khi người dùng chưa đăng nhập. Ở đây ta không gửi dữ liệu đến backend nữa mà sử dụng trực tiếp js để thay đổi cookie. Thông tin giỏ hàng chỉ được lưu lại khi người dùng thanh toán đơn hàng.

## Thêm hình thức thanh toán trực tuyến.

* Tích hợp thanh toán qua ví momo sử dụng code mà momo cung cấp: <https://github.com/momo-wallet/payment/tree/master/python>
* Tài khoản là tài khoản test của momo.
* Thanh toán bằng cách quét mã QR.
* Khi nhấn vào nút thanh toán với người dùng đã đăng nhập thì order sẽ được tạo mới, order cũ của họ sẽ được chuyển sang trạng thái là đã hoàn thành.
* Đối với người chưa đăng nhập thì ta sẽ tạo 1 order mới với trạng thái hoàn thành và lưu tất cả dữ liệu người dùng và dữ liệu trong cookie vào đó.

## Thêm chứ năng đánh giá sản phẩm.

* Tạo một Model mới trong database tên Review
* Chỉ có người đã đăng nhập mới có thể đánh giá sản phẩm
* Trong hàm đánh giá sản phẩm thì nếu đây là lần đầu tiên người dùng đánh giá vô sản phẩm này thì điểm đánh giá sẽ được tính theo công thức sau:

(Điểm \* số lần sản phẩm được đánh giá + điểm của người dùng này) / (số lần đánh giá +1)

* Còn nếu là đánh giá lại thì:

(Điểm \* số lần sản phẩm được đánh giá + điểm của người dùng này – điểm cũ) / (số lần đánh giá)

## Thêm chức năng tìm kiếm sản phẩm.

* Thêm thanh tìm kiếm trên thanh nav trang web.
* Sử dụng filter trong django để lọc sản phẩm

## Thêm chức năng quản lý các đơn hàng đã được đặt.

* Tạo 1 trang bao gồm tất cả các đơn hàng chưa được giao hàng và 1 trang xuất ra thông tin của đơn hàng đó.
* Các hàm liên quan thì chỉ có admin mới có quyền truy cập
* Sau khi admin bấm vào nút hoàn tất đơn hàng thì trạng thái giao hàng của đơn hàng này sẽ chuyển thành đúng

## Deploy trang web lên internet.

* Cài đặt whitenoise
* Cài đặt django gunicorn
* Cài đặt dj-database-url
* Cài đặt psycog2
* Cài đặt Whitenoise
* Cấu hình lại các tệp tĩnh trong setting.py
* Thêm whitenoise vào middleware
* Tạo ứng dụng heroku
* Đẩy toàn bộ code sử dụng git lên heroku
* Migrate dữ liệu trên heroku

# KẾT LUẬN

Chúng em cùng nghiên cứu cách tạo và phát triển một trang web sử dụng ngôn ngữ python, framework django. Từ đó chúng em đã có thể hoàn thành được một trang web cơ bản với đủ mọi yếu tố cần thiết để khách hàng có thể có một trải nghiệm tốt nhất trên website.

Thông qua quá trình phân công và xây dựng trang web chúng em đã học được thêm những kiến thức mới để có thể xây dựng thềm về back-end góp phần tích lũy thêm kinh nghiệm. Trong quá trình làm còn có nhiều thứ còn chưa hợp lý, sai xót, tốn nhiều thời gian để sửa chữa tuy nhiên đó cũng là kinh nghiệm, kiến thức làm hành trang cho công việc tương lai sau này.

Link trang web: <https://shopthuongmai.herokuapp.com/>

**TÀI LIỆU THAM KHẢO**

Cách triển khai ứng dụng Django cho Heroku với Git CLI: <https://viblo.asia/p/cach-trien-khai-ung-dung-django-cho-heroku-voi-git-cli-1VgZvDj15Aw>

Khóa học chính trên youtube: <https://www.youtube.com/watch?v=_ELCMngbM0E&list=PL-51WBLyFTg0omnamUjL1TCVov7yDTRng>

Tham khảo code và sửa lỗi: <https://stackoverflow.com/>

**PHỤ LỤC PHÂN CÔNG NHIỆM VỤ**

|  |  |  |
| --- | --- | --- |
| **STT** | **Thành viên** | **Nhiệm vụ** |
| 1 | Nguyễn Huy Phong | Code python backend, làm phần nội dung và kiểm tra word. |
| 2 | Hồ Sĩ Luân | Tìm hiểu giao diện chỉnh sửa code cho website, tạo dữ liệu, chỉnh sửa code python backend, kiểm tra và hoàn chỉnh word. |