BAN CƠ YẾU CHÍNH PHỦ

PHÂN HIỆU HỌC VIỆN KTMM TẠI TP. HỒ CHÍ MINH

ĐỒ ÁN TỐT NGHIỆP

XÂY DỰNG WEBSITE THƯƠNG MẠI ĐIỆN TỬ AN TOÀN

Ngành: An toàn thông tin

Mã số: 7.48.02.02

Sinh viên thực hiện:

Ngô Quang Sang

Lớp: AT15H

Giảng viên hướng dẫn:

ThS. Nguyễn Thị Kim Oanh

Khoa Công Nghệ Thông Tin - Trường Cao đẳng Nông Nghiệp và Phát triển Nông Thôn Bắc Bộ

Thành phố Hồ Chí Minh, 2023

BAN CƠ YẾU CHÍNH PHỦ

PHÂN HIỆU HỌC VIỆN KTMM TẠI TP. HỒ CHÍ MINH

ĐỒ ÁN TỐT NGHIỆP

XÂY DỰNG WEBSITE THƯƠNG MẠI ĐIỆN TỬ AN TOÀN

Ngành: An toàn thông tin

Mã số: 7.48.02.02

Sinh viên thực hiện:

Ngô Quang Sang

Lớp: AT15H

Giảng viên hướng dẫn:

ThS. Nguyễn Thị Kim Oanh

Khoa Công Nghệ Thông Tin - Trường Cao đẳng Nông Nghiệp và Phát triển Nông Thôn Bắc Bộ

Thành phố Hồ Chí Minh, 2023

LỜI CẢM ƠN

Tôi xin gửi lời cảm ơn chân thành tới các giáo viên hướng dẫn của tôi, ThS. Vũ Thị Vân và ThS. Nguyễn Thị Kim Oanh, vì sự hỗ trợ và định hướng quan trọng của cô trong quá trình phân tích thiết kế kiến trúc hệ thống website thương mại điện tử. Dưới sự chỉ dạy tận tâm của các cô, tôi đã học được rất nhiều kiến thức chuyên môn và nhận được sự truyền cảm hứng về tinh thần trách nhiệm và thái độ làm việc nghiêm túc.

Tôi cũng muốn gửi lời cảm ơn đến tất cả các thầy cô trong học viện Kỹ thuật Mật Mã và khoa an toàn thông tin đã tận tình giảng dạy và truyền đạt những kiến thức và kinh nghiệm quý báu cho tôi trong suốt thời gian học tập tại học viện. Sự quan tâm và tạo mọi điều kiện thuận lợi từ phía các thầy cô cũng đã đóng góp không nhỏ trong quá trình thực hiện đồ án của tôi.

Tôi cũng không thể quên lời cảm ơn đến gia đình, bạn bè và những người đã động viên, đóng góp ý kiến và hỗ trợ tôi trong suốt quá trình học tập, nghiên cứu và hoàn thành đồ án.

Tôi xin gửi lời cảm ơn chân thành tới tất cả mọi người đã đóng góp vào quá trình học tập và hoàn thiện đồ án tốt nghiệp của tôi. Sự giúp đỡ và hỗ trợ từ mọi người là một phần không thể thiếu để tôi có thể đạt được kết quả tốt nhất.

Tôi rất biết ơn sự hướng dẫn và hỗ trợ từ ThS. Vũ Thị Vân và ThS. Nguyễn Thị Kim Oanh và toàn thể những người đã góp phần vào thành công của tôi trong quá trình thực hiện đồ án. Tôi sẽ luôn đánh giá cao sự đóng góp của mọi người và sẽ nỗ lực để áp dụng kiến thức đã học vào thực tế trong tương lai.

Tôi xin chân thành cảm ơn!

Tp. Hồ Chí Minh, ngày 05 tháng 06 năm 2023

Sinh viên thực hiện

Ngô Quang Sang

LỜI CAM ĐOAN

Tôi xin cam đoan bản đồ án này do tôi tự nghiên cứu dưới sự hướng dẫn của giảng viên hướng dẫn ThS. Vũ Thị Vân và ThS. Nguyễn Thị Kim Oanh.

Để hoàn thành đồ án này, tôi chỉ sử dụng những tài liệu đã ghi trong mục tài liệu tham khảo, ngoài ra không sử dụng bất cứ tài liệu nào khác mà không được ghi.

Nếu sai, tôi xin chịu mọi hình thức kỷ luật theo quy định của Học viện.

Tp. Hồ Chí Minh, ngày 05 tháng 06 năm 2023

Sinh viên thực hiện

Ngô Quang Sang

MỤC LỤC

DANH MỤC CÁC KÝ HIỆU, CHỮ VIẾT TẮT

Từ viết tắt	Định nghĩa
TMĐT	Thương mại điện tử
SSL	Secure Socket Layer
CSS	Cascading Style Sheets
HTML	HyperText Markup Language - Ngôn ngữ Đánh dấu Siêu văn bản
RWD	Responsive web design - Thiết kế responsive
URL	Uniform Resource Locator
PCI DSS	Payment Card Industry Data Security Standard
API	Application Programming Interface
DBMS	Database Management System - Hệ quản trị cơ sở dữ liệu
COD	Cash on delivery - Thanh toán khi nhận hàng
SQL	Structured Query Language
CSDL	Cơ sở dữ liệu
AES	Advanced Encryption Standard
JWT	JSON Web Token

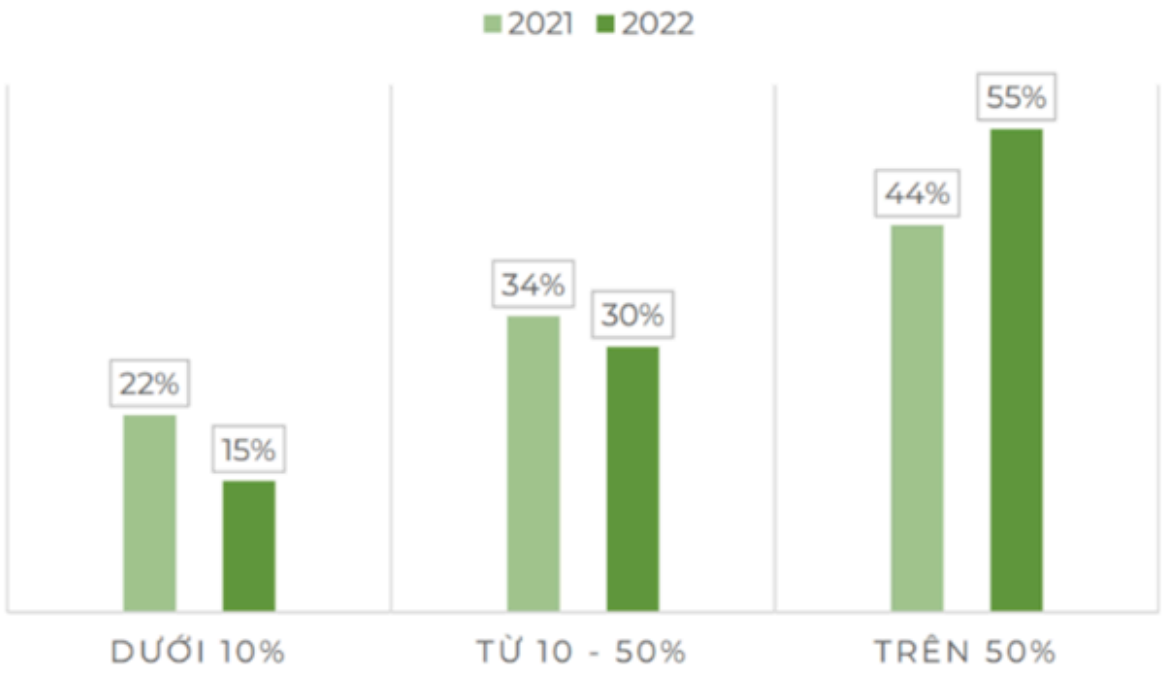
DANH MỤC BẢNG

DANH MỤC HÌNH VẼ, ĐỒ THỊ

MỞ ĐẦU

Số lượng người dùng thương mại điện tử (TMĐT) tăng nhanh: Gần đây, sự phát triển công nghệ mở ra nhiều cơ hội kinh doanh mới, giúp thị trường TMĐT phát triển và thu hút nhiều người dùng hơn.

Theo Hiệp hội Thương mại điện tử Việt Nam, hoạt động kinh doanh trên các sàn TMĐT và mạng xã hội là những nét nổi bật của ngành TMĐT Việt Nam năm 2022 và quý 1/2023. Kết quả khảo sát cho thấy có tới 65% doanh nghiệp đã triển khai hoạt động kinh doanh trên các mạng xã hội. Ngoài ra, số lượng lao động trong doanh nghiệp thường xuyên sử dụng các công cụ như Zalo, WhatsApp, Viber hay Facebook Messenger cũng liên tục tăng qua từng năm. [1]



Hình 1: Người dùng sử dụng các nền tảng mạng xã hội trong 2 năm

Với sự phát triển mang tính toàn cầu của mạng Internet và TMĐT, con người có thể mua bán hàng hoá và dịch vụ thông qua mạng máy tính toàn cầu một cách dễ dàng trong mọi lĩnh vực thương mại rộng lớn. Tuy nhiên đối với các giao dịch mang tính nhạy cảm này cần phải có những cơ chế đảm bảo bảo mật và an toàn vì vậy vấn đề an toàn bảo mật thông tin trong TMĐT là một vấn đề hết sức quan trọng.

Hiện nay vấn đề an toàn bảo mật cho dữ liệu và thanh toán trong TMĐT đã và đang được áp dụng phổ biến và rộng rãi ở Việt Nam và trên phạm vi toàn cầu. Vì thế vấn đề an toàn bảo mật cho dữ liệu và thanh toán đang được nhiều người tập trung nghiên cứu và tìm mọi giải pháp để đảm bảo an toàn bảo mật cho các hệ thống thông tin trên mạng. Tuy nhiên cũng cần phải hiểu rằng không có một hệ thống thông tin nào được bảo mật 100% bất kỳ một hệ thống thông tin nào cũng có những lỗ hổng về bảo mật và an toàn mà chưa được phát hiện ra. Khi giao dịch trực tuyến, người dùng thường cung cấp thông tin cá nhân và tài khoản ngân hàng. Nếu không có biện pháp bảo mật, dữ liệu này có thể bị đánh cắp và lợi dụng để gây hại.

Vấn đề an toàn bảo mật thông tin cho dữ liệu và thanh toán trong TMĐT phải đảm bảo bốn yêu cầu sau đây:

- Đảm bảo tin cậy: Các nội dung thông tin không bị theo dõi hoặc sao chép bởi những thực thể không được uỷ thác.
- Đảm bảo toàn vẹn: Các nội dung thông tin không bị thay đổi bởi những thực thể không được uỷ thác.
- Sự chứng minh xác thực: Không ai có thể tự trá hình như là bên hợp pháp trong quá trình trao đổi thông tin.
- Không thể thoái thác trách nhiệm: Người gửi tin không thể thoái thác về những sự việc và những nội dung thông tin thực tế đã gửi đi.

Mục tiêu chính của đề tài này là tạo ra sản phẩm nhằm tăng cường an toàn và bảo mật trong TMĐT: Triển khai giải pháp xác thực đảm bảo an toàn bảo mật cho dữ liệu và thanh toán giúp tăng cường an toàn và bảo mật cho giao dịch TMĐT, giúp người dùng yên tâm hơn khi giao dịch trực tuyến. Giúp nâng cao uy tín và chất lượng của website TMĐT khi triển khai các giải pháp bảo mật an toàn và đáp ứng các tiêu chuẩn an toàn quốc tế, đó là điểm cộng để nâng cao uy tín và chất lượng của website, thu hút người dùng tin tưởng và sử dụng. Nghiên cứu các kỹ thuật và triển khai các phương pháp mã hóa bảo vệ dữ liệu, chống xâm nhập và đánh cắp dữ liệu và áp dụng các kết quả đã tìm hiểu và nghiên cứu để triển khai hệ thống an toàn bảo mật cho dữ liệu và thanh toán trong TMĐT.

TMĐT đang trở thành lĩnh vực kinh doanh tiềm năng, đóng góp tích cực cho sự phát triển của nền kinh tế và xã hội. Triển khai giải pháp bảo mật an toàn cho TMĐT giúp tạo điều kiện thuận lợi cho sự phát triển của lĩnh vực này, giúp doanh nghiệp TMĐT tăng cường sự tin tưởng của khách hàng và nâng cao hiệu quả kinh doanh. Hiện nay, các quy định về bảo mật thông tin và thanh toán trực tuyến đang được nhiều quốc gia và khu vực áp dụng. Triển khai giải pháp bảo mật an toàn cho TMĐT giúp đáp ứng các tiêu chuẩn và quy định này, giúp website TMĐT tránh được các rủi ro về pháp lý.

Vì vậy, đề tài này có tính cấp thiết và ý nghĩa thực tiễn cao, giúp tăng cường an toàn và bảo mật cho giao dịch TMĐT, đóng góp tích cực cho sự phát triển của lĩnh vực TMĐT và đáp ứng các tiêu chuẩn và quy định của pháp luật.

CHƯƠNG 1. KHẢO SÁT VÀ PHÂN TÍCH CÁC YÊU CẦU CỦA MỘT WEBSITE THƯƠNG MẠI ĐIỆN TỬ (TMĐT)

1.1. Tổng quan về thương mại điện tử

Thương mại điện tử (TMĐT) hiểu một cách đơn giản là hoạt động mua bán sản phẩm hay dịch vụ thông qua Internet và các phương tiện điện tử khác. Các giao dịch này gồm tất cả hoạt động như: mua bán, thanh toán, đặt hàng, quảng cáo và giao hàng … Có nhiều tổ chức lớn trên thế giới đưa ra các định nghĩa khác nhau cho khái niệm của TMĐT.

Theo Ủy ban Kinh tế Liên Hiệp Quốc châu Âu (UNECE): "TMĐT nội địa bao gồm các giao dịch trong nước qua Internet hoặc các mạng máy tính trung gian, trong khi đó, TMĐT quốc tế liên quan đến các giao dịch xuyên biên giới. Các giao dịch này là giao dịch mua/bán hàng hóa hoặc dịch vụ, sau đó, quá trình chuyển giao hàng hóa có thể được thực hiện trực tuyến hoặc thủ công".

Theo Tổ chức Thương mại Thế giới (WTO): "TMĐT bao gồm việc sản xuất, quảng cáo, bán hàng và phân phối sản phẩm được mua bán và thanh toán trên mạng Internet, nhưng được giao nhận một cách hữu hình, cả các sản phẩm giao nhận cũng như những thông tin số hoá thông qua mạng Internet".

Ngày nay ngươi ta còn hiểu khái niệm TMĐT thông thường là tất cả các phương pháp tiến hánh kinh doanh và các quy trình quản trị thông qua các kêng điện tử mà trong đó internet đóng vai trò cơ bản và trong công nghệ thông tin được gọi là điều kiện tiên quyết.

Tuy nhiên, TMĐT không chỉ là kinh doanh sử dụng công nghệ. TMĐT là toàn bộ quá trình kinh doanh được thực hiện bằng điện tử và được thiết kế để giúp hoàn thành mục tiêu kinh doanh. Hai công nghệ chủ chốt để xây dựng và phát triển TMĐT là trao đổi dữ liệu điện tử (EDI) và chuyển tiền điện tử (EFT). Ngày nay, công nghệ chuyển tiền điện tử được ứng dụng để xây dựng các hệ thống thanh toán điện tử.

Website TMĐT là một nền tảng trực tuyến cho phép các doanh nghiệp bán hàng hoặc dịch vụ của mình cho khách hàng thông qua internet. TMĐT cung cấp một kênh bán hàng trực tuyến, mở rộng phạm vi tiếp cận của doanh nghiệp đến toàn bộ thị trường và giúp tăng thu nhập.

1.1.1. Các đặc trưng cơ bản của TMĐT

So với các hoạt động thương mại truyền thống, TMĐT có một số các đặc trưng cơ bản sau:

- Truy cập từ xa.
- Thanh toán trực tuyến.
- Mở cửa 24/7.
- Phạm vi rộng.
- Tiết kiệm chi phí.
- Đa dạng sản phẩm.
- Trải nghiệm người dùng tốt.

1.1.2. Lợi ích của một trang TMĐT

Website TMĐT mang lại nhiều lợi ích cho các doanh nghiệp và khách hàng, bao gồm:

- Mở rộng phạm vi tiếp cận khách hàng.
- Tiết kiệm chi phí.
- Tăng doanh số bán hàng.
- Tăng tính minh bạch trong quản lý kinh doanh.
- Đáp ứng nhu cầu của khách hàng.
- Tăng tính tiện lợi cho khách hàng.
- Tiết kiệm thời gian.
- Hỗ trợ quản lý đơn hàng dễ dàng.

1.1.3. Các hệ thống thanh toán trong TMĐT

Một số hình thức thanh toán điện tử được sử dụng rộng rãi trong các hệ thống thanh toán điện tử được trình bày dưới đây:

- Thanh toán bằng thẻ: Đây là hình thức thanh toán đặc trưng nhất, chiếm tới 90% trong tổng số các giao dịch thanh toán điện tử. Thẻ thanh toán (thẻ chi trả) là một loại thẻ có khả năng thanh toán tiền mua hàng hóa, dịch vụ tại một vài địa điểm, kể cả website mua hàng trực tuyến nếu chấp nhận tiêu dùng bằng thẻ đó. Thẻ có thể dùng để rút tiền mặt trực tiếp từ các ngân hàng hay các máy rút tiền tự động.
- Thanh toán qua cổng thanh toán điện tử: Cổng thanh toán điện tử về bản chất là dịch vụ cho phép khách hàng giao dịch tại các website TMĐT. Cổng thanh toán cung cấp hệ thống kết nối an toàn giữa tài khoản (thẻ, ví điện tử,…) của khách hàng với tài khoản của website bán hàng. Cổng thanh toán điện tử giúp người tiêu dùng và doanh nghiệp thanh toán, nhận tiền trên internet đơn giản, nhanh chóng và an toàn.
- Thanh toán bằng ví điện tử: Ví điện tử là một tài khoản online có thể dùng nhận, chuyển tiền, mua thẻ điện thoại, vé xem phim, thanh toán trực tuyến các loại phí trên internet như tiền điện nước, cước viễn thông, cũng có thể mua hàng online từ các trang TMĐT. Người dùng phải sở hữu thiết bị di động thông minh tích hợp ví điện tử và liên kết với ngân hàng thì mới có thể thanh toán trực tuyến bằng hình thức này.
- Thanh toán qua Mobile Banking: Hình thức này đang dần trở nên phổ biến bởi hầu hết người dùng đều sử hữu một chiếc điện thoại thông minh. Chính vì vậy, khi đi mua sắm, khách hàng không cần phải mang theo tiền mặt, thay vào đó là thanh toán qua điện thoại với dịch vụ Mobile Banking. Hệ thống thanh toán qua điện thoại được xây dựng trên mô hình liên kết giữa ngân hàng, các nhà cung cấp viễn thông, và người dùng.
- Thanh toán qua QR Code: Tiến bộ công nghệ cũng là lý do khiến thanh toán bằng QR Code ngày càng được ưa chuộng. Phương thức thanh toán này khá đơn giản, gọn nhẹ, dễ sử dụng và thân thiện cho người dùng. Tính năng QR Code hiện đang được tích hợp sẵn trên ứng dụng di động của các ngân hàng, các

sản phẩm và dịch vụ của Google như Google Chart hay Google Map, trên bảng hiệu, xe buýt, danh thiếp, tạp chí, website, hàng hóa tại siêu thị, cửa hàng tiện lợi,… Thậm chí là trên một số siêu ứng dụng như VinID của Tập đoàn Vingroup. Người dùng sử dụng camera điện thoại quét mã QR để thực hiện nhanh các giao dịch chuyển khoản, thanh toán hóa đơn, mua hàng. Chỉ với một lần quét, sau vài giây, người dùng đã thanh toán thành công tại các nhà hàng, siêu thị, cửa hàng tiện lợi, taxi, thậm chí là các website TMĐT hay trên bất cứ sản phẩm nào có gắn mã QR mà không cần sử dụng tiền mặt, thẻ, không lo lộ thông tin cá nhân tại các điểm thanh toán.

1.1.4. Quy trình thanh toán điện tử

Các hệ thống thanh toán điện tử triển khai trong thực tế rất đa dạng về hình thức và công nghệ sử dụng.

Hình 2: Quy trình thanh toán điện tử

Trong mô hình ở Hình 2, hệ thống thanh toán điện tử là trung gian kết nối giữa người mua, người bán, thực hiện thanh toán cho các giao dịch dựa trên kết nối với ngân hàng của người mua và người bán:

- Bước 1 - Thẻ tín dụng: Các khách hàng có thẻ tín dụng do ngân hàng phát hành với hạn mức tín dụng và số dư có sẵn.
- Bước 2 - Đặt hàng: Các khách hàng đến thăm một trang web hoặc cửa hàng trực tuyến sử dụng trình duyệt web tiêu chuẩn và bắt đầu mua sắm và thêm (các) sản phẩm vào giỏ hàng của mình. Sau khi kiểm tra, người mua được yêu cầu để gửi thông tin thẻ tín dụng của mình, ngày hết hạn, địa chỉ thanh toán. Sau

đó, người mua cũng chọn phương thức vận chuyển cho ví dụ và sau đó nhấn vào nút gửi để bắt đầu giao dịch. Các thông tin này sau đó được chuyển đến cửa hàng trực tuyến của thương gia nơi mà các dịch vụ thanh toán bên ngoài được thiết lập. Các dịch vụ thanh toán bên ngoài nhận được thông tin được mã hóa từ các cửa hàng trực tuyến, thực hiện một kiểm tra gian lận, và sau đó bắt đầu quá trình giao tiếp thông tin thanh toán và số tiền mua hàng cho các bộ xử lý của bên thứ ba.

- Bước 3 - Yêu cầu xác nhận: Dịch vụ dịch thanh toán mã hóa thông tin mua hàng hoặc dữ liệu và truyền nó cho các bộ xử lý của bên thứ ba, người sẽ chuyển thông tin hoặc dữ liệu hơn nữa để các hiệp hội thẻ hoặc thẻ phát hành cho phép và xác minh
- Bước 4 - Đáp trả xác thực: Các tổ chức tài chính phát hành xác minh thông tin thẻ tín dụng và xác định xem khách hàng có đủ tín dụng để thanh toán tiền mua. Một số quyền được tạo ra và tín dụng là giảm lượng có thẩm quyền. Nếu nó để xảy ra rằng các thông tin thẻ tín dụng là không đúng hoặc nếu không có đủ tín dụng có sẵn, sau đó nhắn giảm giao dịch được tạo ra. Trong khoảng thời gian ngắn này của thời gian, các ngân hàng phát hành cũng thực hiện các hoạt động khác như dịch vụ xác minh địa chỉ, nơi các thông tin thanh toán đã nhập trực tuyến được so sánh với các mục trong cơ sở dữ liệu của ngân hàng phát hành - đây là phần xác thực. Sau đó, một thông báo uỷ quyền được trả lại cho các hiệp hội thẻ và chuyển tiếp đến các bộ xử lý của bên thứ ba.
- Bước 5 - Thông báo cho bên bán: Bộ xử lý của bên thứ ba nhận được tin nhắn uỷ quyền và các thông tin cần thiết khác từ các hiệp hội thẻ hoặc tổ chức phát hành và khởi tạo quá trình truyền đạt thông điệp uỷ quyền cho các thương gia. Các bộ vi xử lý của bên thứ ba mã hóa thông điệp uỷ quyền và truyền các thông tin mã hóa cho máy chủ thương mại an toàn của bên bán.
- Bước 6 - Thông báo từ bên bán: máy chủ của bên bán thu được các thông tin và được lập trình để gửi ngay cho chính người mua hoặc tin nhắn cho chủ thẻ/khách hàng. Thông thường khi thẻ tín dụng bị từ chối, một số thông tin cần thiết như một gợi ý để kiểm tra tính chính xác của các thông tin được cung cấp hoặc sử dụng một thẻ tín dụng khác để gửi lên. Ngay sau khi khách hàng nhận

được thông tin này sẽ đồng ý chấp thuận một giao dịch, cùng một lúc nhận được một số xác nhận. Nó chỉ mất một vài giây từ thời điểm khách hàng nhấn vào nút mua cho đến khi khách hàng nhận được tin nhắn phản hồi trở lại. Các quá trình cấp phép thường mất một vài giây, tùy thuộc vào ứng dụng của bên bán thanh toán và thủ tục cũng như lưu lượng Internet và các yếu tố khác.

- Bước 7 - Hoàn thành: Bên bán bắt đầu quá trình thực hiện lệnh của khách hàng với các sản phẩm/dịch vụ thích hợp.
- Bước 8 - Yêu cầu giải quyết: Các bên bán biên soạn một loạt các đơn đặt hàng đã được hoàn thành và bắt đầu quá trình truyền tải hàng loạt các bộ xử lý của bên thứ ba để giải quyết. Bên bán đầu tiên truyền hàng loạt dịch vụ thanh toán của mình để mã hoá thông tin mua hàng và truyền các thông tin mã hóa cho các bộ vi xử lý của bên thứ ba. Các bộ xử lý của bên thứ ba nhận được thông tin này và sẽ gửi các hướng dẫn giải quyết cho tổ chức tài chính của mình để chuyển số tiền từ tài khoản của chủ thẻ vào tài khoản của thương gia.
- Bước 9 - Giải quyết: Đối với mỗi giao dịch thẻ tín dụng trong hàng loạt, các tổ chức tài chính thích hợp được ghi nợ và thẻ tín dụng của chủ thẻ được cập nhật. Các ngân hàng bên mua nhận tiền và tiền được gửi vào tài khoản ngân hàng của bên bán.
- Bước 10 - Đáp ứng giải quyết: Bên bán nhận được thông báo rằng tiền đã được gửi vào tài khoản ngân hàng của mình. Trên cơ sở đó, các thương gia nhận được báo cáo rằng ông có thể sử dụng để hòa giải với yêu cầu giải quyết hàng loạt của mình với hoạt động tiền gửi của mình.
- Bước 11 - Quỹ có sẵn: Khoảng cách giữa các cấp của thương gia được yêu cầu thanh toán, chuyển tiền và các quỹ sẵn có thể mất đến vài ngày, tùy thuộc vào các ngân hàng phát hành, các ngân hàng mua lại và các bộ xử lý của bên thứ ba. Chu kỳ thời gian giải quyết là thực sự bị ảnh hưởng bởi thời gian nắm giữ ngân hàng mua lại tiền gửi, cũng như các thủ tục khác và chính sách được thiết lập bởi các ngân hàng mua lại và xử lý của bên thứ ba.

1.2. Mô tả bài toán xây dựng website TMĐT

Công ty ABC kinh doanh về lĩnh vực may mặc muốn xây dựng hệ thống website TMĐT với các yêu cầu:

- Đăng ký và đăng nhập: Người dùng có thể đăng ký tài khoản mới và đăng nhập vào hệ thống. Đăng ký tài khoản dễ dàng.
- Xem sản phẩm: Người dùng có thể xem thông tin chi tiết về các sản phẩm đang được bán trên website.
- Thêm sản phẩm vào giỏ hàng: Người dùng có thể thêm sản phẩm vào giỏ hàng của mình.
- Quản lý giỏ hàng: Người dùng có thể chỉnh sửa số lượng sản phẩm trong giỏ hàng, xóa sản phẩm khỏi giỏ hàng, hoặc xem tổng giá trị đơn hàng.
- Đặt hàng: Người dùng có thể đặt hàng bằng cách cung cấp thông tin về địa chỉ giao hàng, phương thức thanh toán và các chi tiết khác liên quan đến đơn hàng.
- Theo dõi đơn hàng: Người dùng có thể theo dõi trạng thái và thông tin liên quan đến đơn hàng đã đặt.
- Thanh toán: Người dùng có thể chọn phương thức thanh toán và thực hiện thanh toán cho đơn hàng. Tích hợp nhiều dịch vụ thanh toán.
- Giao hàng: Website hỗ trợ các nhà cung cấp dịch vụ giao hàng để vận chuyển các đơn hàng đến địa chỉ giao hàng của người dùng.
- Quản lý người dùng: Website cho phép quản lý thông tin người dùng, bao gồm cập nhật thông tin cá nhân và quản lý địa chỉ giao hàng.
- Quản lý sản phẩm: Website cho phép quản lý thông tin về sản phẩm, bao gồm thêm, sửa, xóa và hiển thị sản phẩm.
- Quản lý đơn hàng: Website cho phép quản lý thông tin về các đơn hàng, bao gồm xem danh sách đơn hàng, cập nhật trạng thái và xem chi tiết từng đơn hàng.
- Giao diện quản lý đơn giản dễ sử dụng, giao diện người dùng dễ nhìn, đặt hàng nhanh chóng.
- Đảm bảo an toàn và bảo mật cho dữ liệu.
- Đảm bảo an toàn cho tính năng thanh toán đơn hàng.

1.3. Khảo sát các nghiệp vụ của một website TMĐT

Một website thương mại điện tử có thể có nhiều nghiệp vụ khác nhau, nhưng dưới đây là một số nghiệp vụ quan trọng và phổ biến mà một website thương mại điện tử thường thực hiện:

- Quản lý sản phẩm: Nghiệp vụ này liên quan đến việc quản lý thông tin về các sản phẩm được bán trên website. Bao gồm việc thêm, sửa, xóa sản phẩm, cập nhật thông tin sản phẩm, quản lý danh mục sản phẩm, hình ảnh, mô tả, giá cả và số lượng hàng tồn kho.
- Quản lý đơn hàng: Nghiệp vụ này bao gồm quản lý quá trình đặt hàng từ khách hàng, xử lý đơn hàng, lưu trữ thông tin đơn hàng, cập nhật trạng thái đơn hàng (đã xác nhận, đang vận chuyển, đã giao hàng, hủy đơn hàng, v.v.), cung cấp thông tin vận chuyển và theo dõi đơn hàng.
- Thanh toán và giao dịch: Nghiệp vụ này liên quan đến việc xử lý thanh toán trực tuyến và giao dịch với khách hàng. Bao gồm tích hợp các cổng thanh toán, cung cấp các phương thức thanh toán an toàn và tiện lợi (thẻ tín dụng, chuyển khoản ngân hàng, ví điện tử, v.v.), xử lý giao dịch thành công và cung cấp thông tin về giao dịch cho khách hàng và nhà cung cấp.
- Quản lý tài khoản khách hàng: Nghiệp vụ này liên quan đến quản lý thông tin cá nhân của khách hàng, bao gồm việc đăng ký tài khoản, đăng nhập, cập nhật thông tin cá nhân, quản lý địa chỉ giao hàng và thông tin thanh toán, theo dõi lịch sử mua hàng và đánh giá sản phẩm.
- Quảng cáo và khuyến mãi: Nghiệp vụ này liên quan đến việc quảng bá sản phẩm và dịch vụ của website, thiết kế và triển khai các chiến dịch quảng cáo, cung cấp mã giảm giá, khuyến mãi và chương trình thưởng cho khách hàng, theo dõi hiệu quả quảng cáo và khuyến mãi.
- Tương tác khách hàng: Nghiệp vụ này liên quan đến việc tương tác với khách hàng qua các kênh như email, chat trực tuyến, điện thoại hoặc mạng xã hội. Bao gồm hỗ trợ khách hàng, giải đáp thắc mắc, xử lý khiếu nại và cung cấp hỗ trợ sau bán hàng.

- Quản lý kho hàng: Nghiệp vụ này liên quan đến việc quản lý và kiểm soát hàng tồn kho, cập nhật số lượng hàng hóa có sẵn, quản lý nhập xuất kho, tổ chức vận chuyển và lưu trữ thông tin về kho hàng.
- Phân tích dữ liệu và báo cáo: Nghiệp vụ này liên quan đến việc thu thập, phân tích và báo cáo dữ liệu về hoạt động kinh doanh của website thương mại điện tử. Bao gồm thông tin về lượt truy cập, doanh số bán hàng, hành vi khách hàng, hiệu quả quảng cáo và khuyến mãi, v.v. để đưa ra quyết định kinh doanh và cải thiện hiệu suất.

Những nghiệp vụ trên chỉ là một phần trong tổng thể của một website thương mại điện tử, và còn nhiều nghiệp vụ khác như quản lý đánh giá và nhận xét sản phẩm, tích hợp công cụ tìm kiếm, xây dựng cộng đồng khách hàng, v.v. Tùy thuộc vào quy mô và phạm vi của website, các nghiệp vụ có thể được điều chỉnh và mở rộng để phù hợp với yêu cầu và mục tiêu kinh doanh của doanh nghiệp.

1.4. Các yêu cầu chức năng của một website TMĐT

Một website thương mại điện tử cần phải đáp ứng một số yêu cầu chức năng để cung cấp trải nghiệm mua sắm trực tuyến thuận tiện, hiệu quả cho người dùng và cung cấp công cụ quản lý và phát triển hiệu quả cho người bán. Dưới đây là một số yêu cầu chức năng quan trọng của một website TMĐT:

- Đăng ký và đăng nhập: Cho phép người dùng đăng ký tài khoản mới, cung cấp thông tin cá nhân cần thiết và đăng nhập vào tài khoản đã đăng ký để tiếp tục quá trình mua sắm.
- Tìm kiếm và duyệt sản phẩm: Cung cấp chức năng tìm kiếm để người dùng có thể tìm kiếm sản phẩm theo từ khóa, danh mục, giá cả, thương hiệu, v.v. Đồng thời, cung cấp khả năng duyệt sản phẩm theo danh mục, thương hiệu, sản phẩm mới nhất, sản phẩm bán chạy, v.v.
- Chi tiết sản phẩm: Cung cấp thông tin chi tiết về sản phẩm bao gồm hình ảnh, mô tả, thông số kỹ thuật, giá cả, đánh giá và nhận xét từ người dùng khác.
- Giỏ hàng và thanh toán: Cho phép người dùng thêm sản phẩm vào giỏ hàng, cập nhật số lượng sản phẩm, tính tổng giá trị đơn hàng, chọn phương thức

thanh toán và cung cấp thông tin thanh toán cần thiết để hoàn tất quá trình mua hàng.

- Quản lý đơn hàng: Cung cấp chức năng cho người dùng theo dõi và quản lý đơn hàng, bao gồm xem trạng thái đơn hàng, lịch sử đơn hàng, in hóa đơn và yêu cầu hỗ trợ sau bán hàng.
- Quản lý tài khoản: Cho phép người dùng cập nhật thông tin cá nhân, địa chỉ giao hàng, thông tin thanh toán và thay đổi mật khẩu.
- Đánh giá và nhận xét: Cho phép người dùng đánh giá và viết nhận xét về sản phẩm đã mua để chia sẻ kinh nghiệm và đánh giá với người dùng khác.
- Quản lý khuyến mãi và mã giảm giá: Cung cấp chức năng quản lý các chương trình khuyến mãi, mã giảm giá và ưu đãi đặc biệt, cho phép người dùng áp dụng mã giảm giá khi thanh toán.
- Giao hàng và vận chuyển: Cung cấp thông tin về phương thức giao hàng, thời gian giao hàng dự kiến và tính phí vận chuyển. Cho phép người dùng chọn phương thức giao hàng và cập nhật địa chỉ giao hàng.
- Hỗ trợ khách hàng: Cung cấp kênh liên hệ và hỗ trợ khách hàng trực tuyến để giải đáp thắc mắc, xử lý khiếu nại và cung cấp hỗ trợ sau bán hàng.
- Tích hợp thanh toán trực tuyến: Hỗ trợ các phương thức thanh toán trực tuyến an toàn và thuận tiện như thẻ tín dụng, chuyển khoản ngân hàng, ví điện tử, v.v.
- Quản lý báo cáo và thống kê: Cung cấp chức năng phân tích dữ liệu và báo cáo về doanh số bán hàng, lượt truy cập, đánh giá sản phẩm, v.v. để giúp doanh nghiệp đánh giá hiệu suất kinh doanh và đưa ra quyết định phát triển.

Những yêu cầu chức năng này sẽ tùy thuộc vào quy mô và mục tiêu kinh doanh của mỗi website thương mại điện tử. Việc phân tích và đáp ứng đúng các yêu cầu chức năng sẽ giúp đảm bảo trải nghiệm mua sắm tốt nhất cho người dùng và nâng cao hiệu quả kinh doanh của doanh nghiệp.

1.5. Khảo sát, phân tích các yêu cầu an toàn của một website TMĐT

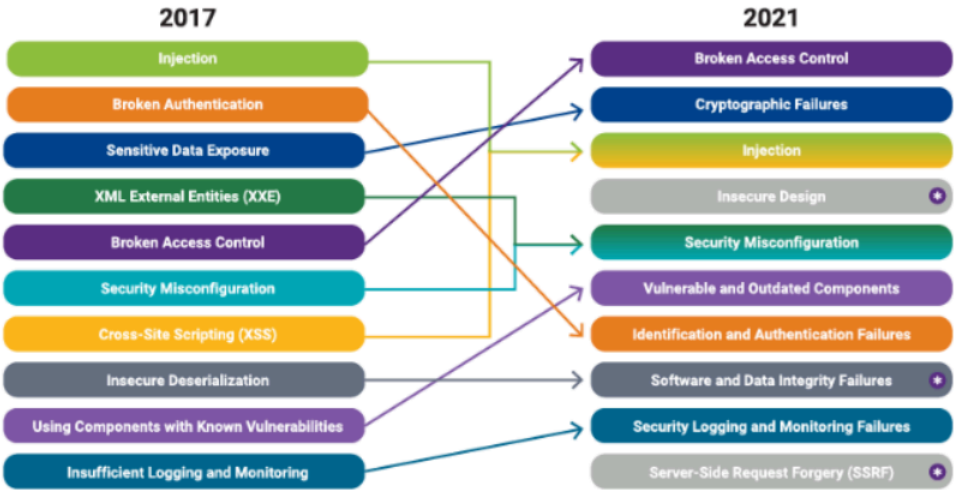
1.5.1. Các lỗ hổng an toàn phổ biến trong website TMĐT

OWASP (Open Web Application Security Project) là một tổ chức phi lợi nhuận quốc tế tập trung vào việc nghiên cứu và phân tích các vấn đề về bảo mật ứng dụng web. OWASP cung cấp nhiều tài liệu, công cụ và khóa học để giúp các nhà phát triển phát hiện và khắc phục các lỗ hổng bảo mật trong ứng dụng web.

OWASP cũng tạo ra danh sách "OWASP Top 10" để tập trung vào 10 lỗ hổng bảo mật phổ biến nhất trong ứng dụng web. Danh sách này được cập nhật định kỳ và dùng làm hướng dẫn cho việc phát triển ứng dụng an toàn. Các phiên bản của OWASP Top 10 từng được phát hành vào các năm khác nhau và được điều chỉnh để phản ánh các mối đe dọa bảo mật mới nhất.

Theo tổ chức OWASP, 10 lỗ hổng ứng dụng web phổ biến nhất hiện nay bao gồm: [2]

- Broken Access Control
- Cryptographic Failures
- Injection
- Insecure Design (Lỗ hổng mới cập nhật)
- Security Misconfiguration
- Vulnerable and Outdated Components
- Identification and Authentication Failures
- Software and Data Integrity Failures (Lỗ hổng mới cập nhật)
- Security Logging and Monitoring Failures
- Server-Side Request Forgery (SSRF) (Lỗ hổng mới cập nhật)



Hình 3: Danh sách lỗ hổng Top 10 OWASP 2021

OWASP Top 10 được cập nhật lại sau một khoảng thời gian tương đối, tùy thuộc vào sự phát triển và xu hướng của các lỗ hổng bảo mật trong ứng dụng web. Thông thường, OWASP Top 10 được cập nhật mỗi 3 - 4 năm. Danh sách OWASP Top 10 trước đây đã có các phiên bản cho các năm 2003, 2004, 2007, 2010, 2013 và 2017.

1.5.1.1. Lỗ hổng Broken Access Control

Kiểm soát truy cập là sự kiểm soát người dùng không cho phép họ thực thi những hành động bên ngoài quyền hạn. Các lỗi thường dẫn đến tiết lộ thông tin trái phép, sửa đổi hoặc phá hủy tất cả dữ liệu hoặc thực hiện chức năng ngoài giới hạn của người dùng.

Các phương pháp phổ biến:

- Sửa đổi URL.
- Sửa đổi thông tin nhận dạng để truy cập tài khoản người khác (IDOR).
- Leo thang đặc quyền.

Biện pháp phòng chống:

- Ngoại trừ tài nguyên công cộng, còn lại từ chối theo mặc định.
- Xác thực người dùng khi học quay lại ứng dụng.
- Kiểm tra quyền tại thời điểm người dùng cố gắng thực hiện hành động.

Ví dụ: Kẻ tấn công chỉ cần buộc các trình duyệt đến các URL mục tiêu. Quyền quản trị được yêu cầu để truy cập vào trang quản trị: `https://example.com/app/getappInfo` => `https://example.com/app/admin`.

1.5.1.2. Lỗ hổng Cryptographic Failures

Bảo mật thông tin nhạy cảm bằng cách mã hóa thông tin theo các cách khác nhau, nhưng nếu cách mã hóa đó kẻ tấn công có thể giải mã được hay là cách thức giải mã không đảm bảo an toàn bản rõ thì những thông tin nhạy cảm đó sẽ bị rò rỉ ra ngoài.

Các phương pháp phổ biến:

- Sử dụng những giao thức truyền dữ liệu dạng rõ như HTTP, FTP,…
- Sử dụng những mã hóa đã cũ hoặc yếu.
- Sử dụng những hàm băm không dùng nữa như MD5, SHA1.
- Khóa bí mật dễ đoán.
- Chuỗi mã hóa không được xác thực.

Biện pháp phòng chống:

- Không sử dụng những giao thức đã cũ như FTP, SMTP,… để vận chuyển dữ liệu nhạy cảm.
- Đảm bảo các thuật toán mã hóa đạt tiêu chuẩn mạnh mẽ.
- Mã hóa dữ liệu trên đường truyền bằng TLS, HTTPS.
- Lưu trữ password bằng các hàm băm mạnh như Argon2, scrypt, bcrypt,…
- Luôn sử dụng mã hóa được xác thực thay vì chỉ mã hóa.

Ví dụ: Một trang web không sử dụng TLS cho tất cả các trang hoặc hỗ trợ mã hóa yếu. Kẻ tấn công giám sát lưu lượng mạng (như tại một mạng không dây không an toàn), hạ cấp các kết nối từ HTTPS xuống HTTP, chặn các yêu cầu và đánh cắp cookie phiên của người dùng. Sau đó, kẻ tấn công phát lại cookie này và chiếm quyền điều khiển phiên của người dùng, truy cập hoặc sửa đổi dữ liệu cá nhân của người dùng. Thay vì những điều trên, họ có thể thay đổi tất cả dữ liệu được vận chuyển, ví dụ như người nhận chuyển tiền.

1.5.1.3. Lỗ hổng SQL Injection

Lỗi bảo mật SQL Injection là một trong những lỗi phổ biến nhất trong các website TMĐT. Đây là lỗi bảo mật cho phép kẻ tấn công thực hiện các cuộc tấn công vào cơ sở dữ liệu của trang web bằng cách chèn các câu lệnh SQL độc hại vào các trường đầu vào trên trang web.

Khi khai thác lỗi SQL Injection, kẻ tấn công có thể truy xuất và thay đổi dữ liệu trong cơ sở dữ liệu của trang web, thực hiện các hoạt động xóa hoặc thêm mới dữ liệu, và thậm chí kiểm soát toàn bộ trang web.

Với những biện pháp trên, trang web TMĐT sẽ giảm thiểu được rủi ro bị tấn công SQL Injection và đảm bảo an toàn cho khách hàng trong quá trình giao dịch mua bán sản phẩm trên trang web.

Ví dụ, trong một hệ thống với 1000 đầu vào, lọc thành công 999 đầu vào là không đủ vì điều này vẫn để lại một phần giống như "gót chân Asin", có thể phá hoại hệ thống bất cứ lúc nào. Ta có thể cho rằng đưa kết quả truy vấn SQL vào truy vấn khác là một ý tưởng hay vì cơ sở dữ liệu là đáng tin cậy. Nhưng thật không may vì đầu vào có thể gián tiếp đến từ những kẻ có ý đồ xấu. Đây được gọi là lỗi Second Order SQL Injection.

Việc lọc dữ liệu khá khó vì thế nên sử dụng các chức năng lọc có sẵn trong framework. Các tính năng này đã được chứng minh sẽ thực hiện việc kiểm tra một cách kỹ lưỡng. Vì thế ta nên cân nhắc sử dụng các framework vì đây là một trong các cách hiệu quả để bảo vệ máy chủ.

1.5.1.4. Lỗ hổng Insecure Design (Thiết Kế Không An Toàn)

Thiết kế an toàn là phân tích các giả định và điều kiện cho các dòng dự kiến đảm bảo chính xác, tránh trường hợp không mong muốn và có hành vi phù hợp với từng trường hợp. Đảm bảo kết quả được ghi lại trong nhật kí của người dùng. Học hỏi từ những sai lầm và đưa ra những cải tiến thích hợp.

Biện pháp phòng chống:

- Thiết lập sử dụng những thư viện mẫu thiết kế an toàn.
- Kiểm tra tính hợp lí ở mỗi cấp ứng dụng.

- Tách các lớp phần trên hệ thống và các lớp mạng.
- Hạn chế tiêu thụ tài nguyên người dùng hoặc dịch vụ.

Ví dụ: Một rạp chiếu phim cho phép đặt chỗ theo nhóm tối đa 15 người trước khi đặt tiền cọc, một kẻ tấn công có thể chạy lệnh để đặt tất cả các chỗ trong rạp sau đó dừng lại ở bước đặt cọc, gây tổn thất lớn về kinh tế.

1.5.1.5. Lỗ hổng Security Misconfiguration

Nếu Insecure Design thuộc về phần thiết kế thì Security Misconfiguration thuộc về phần triển khai. Những lỗi phổ biến thường xảy ra:

- Các tính năng không cần thiết được bật như các port, service, account,…
- Thiếu việc tăng cường bảo mật cho từng phần của ứng dụng.
- Các tài khoản và mật khẩu vẫn để mặc định không thay đổi.
- Phần mềm đã lỗi thời.

Trong thực tế, máy chủ website và các ứng dụng đa số bị cấu hình sai. Có lẽ do một vài sai sót như:

- Chạy ứng dụng khi chế độ debug được bật.
- Directory listing.
- Sử dụng phần mềm lỗi thời (WordPress plugin, PhpMyAdmin cũ).
- Cài đặt các dịch vụ không cần thiết.
- Không thay đổi default key hoặc mật khẩu.
- Trả về lỗi xử lý thông tin cho kẻ tấn công lợi dụng để tấn công, chẳng hạn như stack traces.

Biện pháp phòng chống:

- Loại bỏ những tài nguyên, tính năng không cần thiết.
- Cung cấp sự hiệu quả và an toàn giữa các thành phần.
- Liên tục cập nhật những phiên bản mới nhất.

Ví dụ: Danh sách thư mục không bị tắt trên máy chủ. Kẻ tấn công phát hiện ra chúng có thể liệt kê các thư mục một cách đơn giản. Điều này có thể dẫn đến

kẻ tấn công dịch ngược lại đoạn code và là tiềm ẩn rất lớn cho nhiều mối nguy hiểm khác.

1.5.2. Các yêu cầu an toàn của một website TMĐT

Một website thương mại điện tử cần đáp ứng nhiều yêu cầu an toàn để bảo vệ thông tin cá nhân của người dùng và người bán, đảm bảo giao dịch an toàn và tránh các rủi ro liên quan đến bảo mật. Dưới đây là một số yêu cầu an toàn quan trọng của một website TMĐT: [3]

- Xác thực và quản lý truy cập: Cung cấp hệ thống xác thực đáng tin cậy để đảm bảo rằng chỉ người dùng hợp lệ mới có thể truy cập vào tài khoản và thông tin cá nhân. Đồng thời, quản lý quyền truy cập để giới hạn quyền truy cập chỉ cho những người được ủy quyền.
- Bảo vệ thông tin cá nhân: Đảm bảo rằng thông tin cá nhân của người dùng được bảo vệ an toàn và không bị truy cập, sử dụng hoặc tiết lộ trái phép. Áp dụng các biện pháp bảo mật như mã hóa dữ liệu, sử dụng giao thức an toàn (SSL/TLS) cho việc truyền tải thông tin và tuân thủ quy định về bảo vệ dữ liệu cá nhân.
- Bảo mật giao dịch: Đảm bảo rằng quá trình thanh toán và giao dịch trên website được bảo mật. Sử dụng giao thức mã hóa SSL/TLS để bảo vệ thông tin thanh toán, hạn chế lưu trữ thông tin thẻ tín dụng và áp dụng các biện pháp kiểm tra giao dịch an toàn như xác thực hai yếu tố (2FA).
- Quản lý mã độc và tấn công: Đảm bảo hệ thống website không bị tấn công bởi các loại mã độc, phần mềm độc hại và các cuộc tấn công mạng khác. Áp dụng các biện pháp bảo vệ bảo mật cơ bản như cập nhật hệ điều hành và phần mềm định kỳ, kiểm tra lỗ hổng bảo mật, sử dụng tường lửa (firewall), và các biện pháp phòng ngừa tấn công.
- Quản lý rủi ro liên quan đến dữ liệu: Đảm bảo rằng dữ liệu người dùng và dữ liệu liên quan đến giao dịch được lưu trữ, sao lưu và phục hồi một cách an toàn. Thực hiện các biện pháp bảo vệ dữ liệu như mã hóa dữ liệu, sao lưu định kỳ và đảm bảo tính toàn vẹn của dữ liệu.

- Quản lý lỗ hổng bảo mật: Thực hiện việc theo dõi, phát hiện và xử lý lỗ hổng bảo mật trong hệ thống website. Cập nhật và vá lỗi phần mềm định kỳ, thực hiện quản lý rủi ro bảo mật để ngăn chặn và giảm thiểu các mối đe dọa tiềm năng.
- Tuân thủ quy định pháp luật: Đảm bảo rằng website tuân thủ các quy định và quyền riêng tư liên quan đến bảo vệ thông tin cá nhân, bảo mật giao dịch và quyền lợi của người dùng. Điều này bao gồm việc tuân thủ các quy định như GDPR (Quy định chung về bảo vệ dữ liệu) và các quy định pháp luật liên quan đến thương mại điện tử.
- Giám sát và phản ứng sự cố: Thiết lập hệ thống giám sát liên tục để theo dõi các hoạt động bất thường và phản ứng kịp thời đối với các sự cố bảo mật. Đưa ra các biện pháp phòng ngừa và ứng phó sự cố để đảm bảo rằng website được bảo vệ một cách liên tục.

Những yêu cầu an toàn này đặc biệt quan trọng trong lĩnh vực thương mại điện tử, vì thông tin cá nhân và giao dịch của khách hàng là tài sản quý giá và phải được bảo vệ một cách tốt nhất. Sự tuân thủ và triển khai hiệu quả các yêu cầu an toàn này sẽ giúp xây dựng niềm tin và tăng cường sự thụ động của người dùng trong việc sử dụng website thương mại điện tử.

1.6. Kết chương

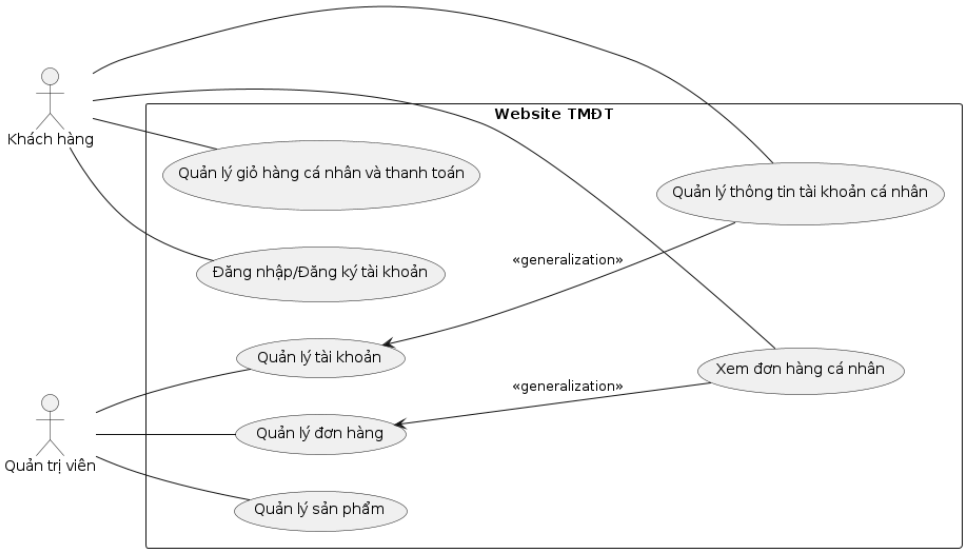
Trong chương 1 đã tìm hiểu về TMĐT và website TMĐT, tìm hiểu các lỗi bảo mật phổ biến mới nhất hiện nay. Từ những phần tìm hiểu này cho thấy việc xây dựng 1 trang website TMĐT an toàn với doanh nghiệp cũng như khách hàng sử dụng là vô cùng quan trọng. Phần tiếp theo của đồ án sẽ nói về phân tích và thiết kế website TMĐT xây dựng dưa trên việc khảo sát và xác định yêu cầu ở Chương 1.

CHƯƠNG 2. PHÂN TÍCH VÀ THIẾT KẾ MÔ HÌNH WEBSITE TMĐT

2.1. Phân tích các yêu cầu nghiệp vụ và chức năng

Các tác nhân trong website:

- Khách hàng.
- Quản trị viên.

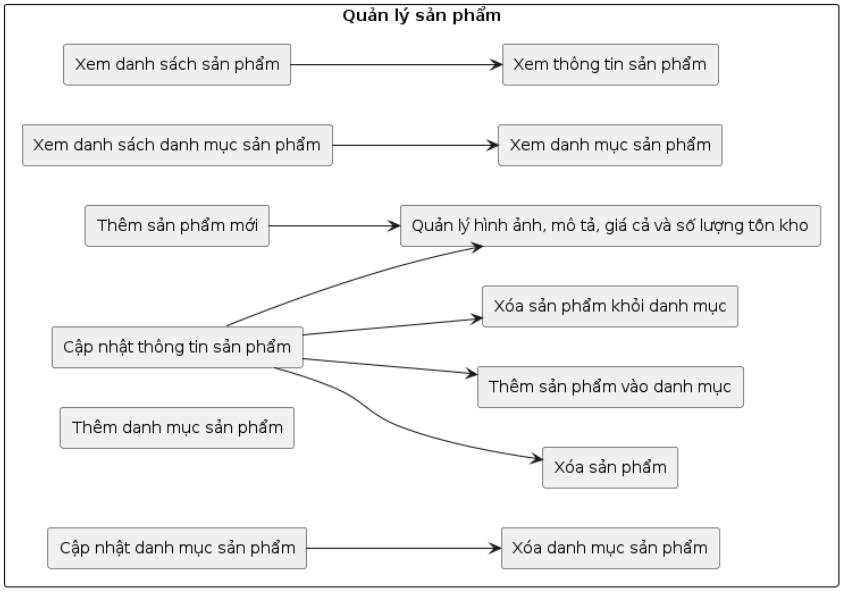


Hình 4: Biểu đồ usecase tổng quát của hệ thống

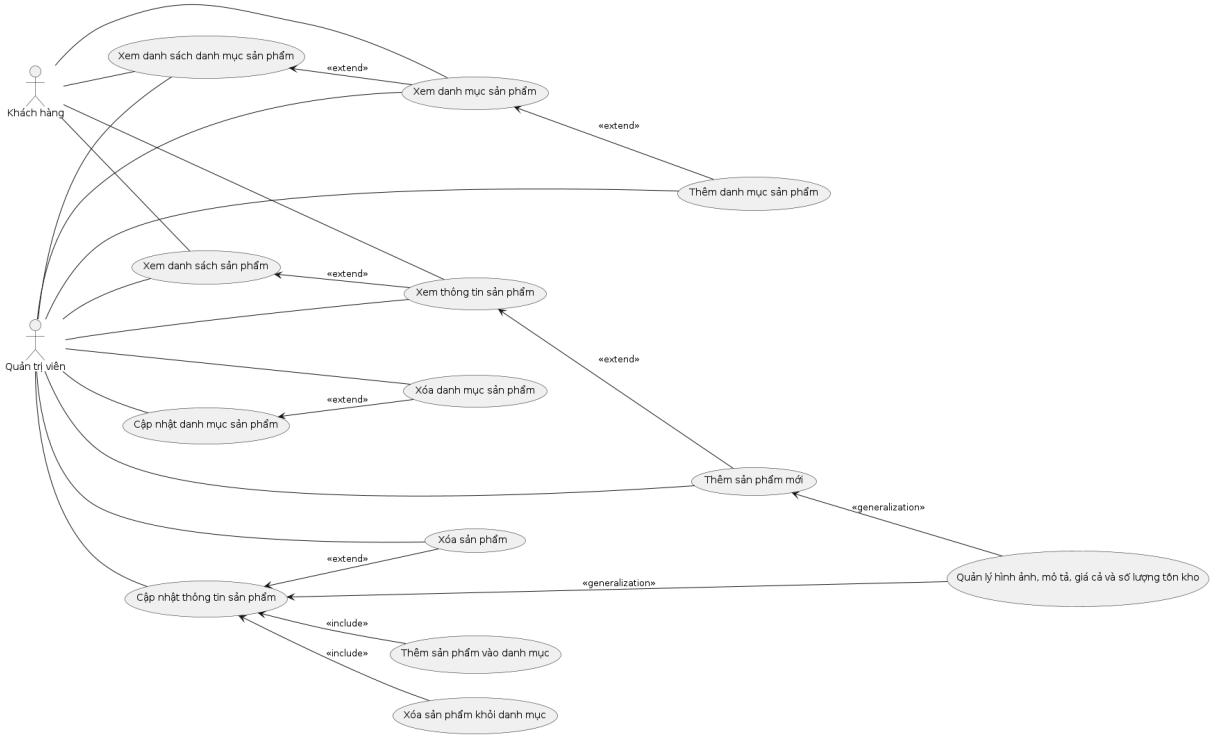
2.1.1. Chức năng quản lý sản phẩm

Danh sách các chức năng con của chức năng quản lý sản phẩm:

- Thêm sản phẩm.
- Xem thông tin sản phẩm.
- Sửa thông tin sản phẩm.
- Xóa sản phẩm.
- Quản lý danh mục sản phẩm.



Hình 5: Biểu đồ phân rã chức năng quản lý sản phẩm



Hình 6: Biểu đồ usecase cho chức năng quản lý sản phẩm

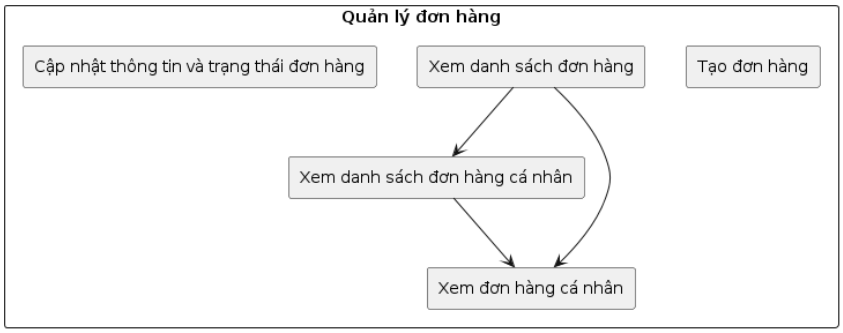
2.1.2. Chức năng quản lý đơn hàng

Chức năng quản lý đơn hàng cho phép khách hàng xem lại các đơn hàng đã đặt trước đó và theo dõi trạng thái của từng đơn hàng. Khách hàng có thể xem chi

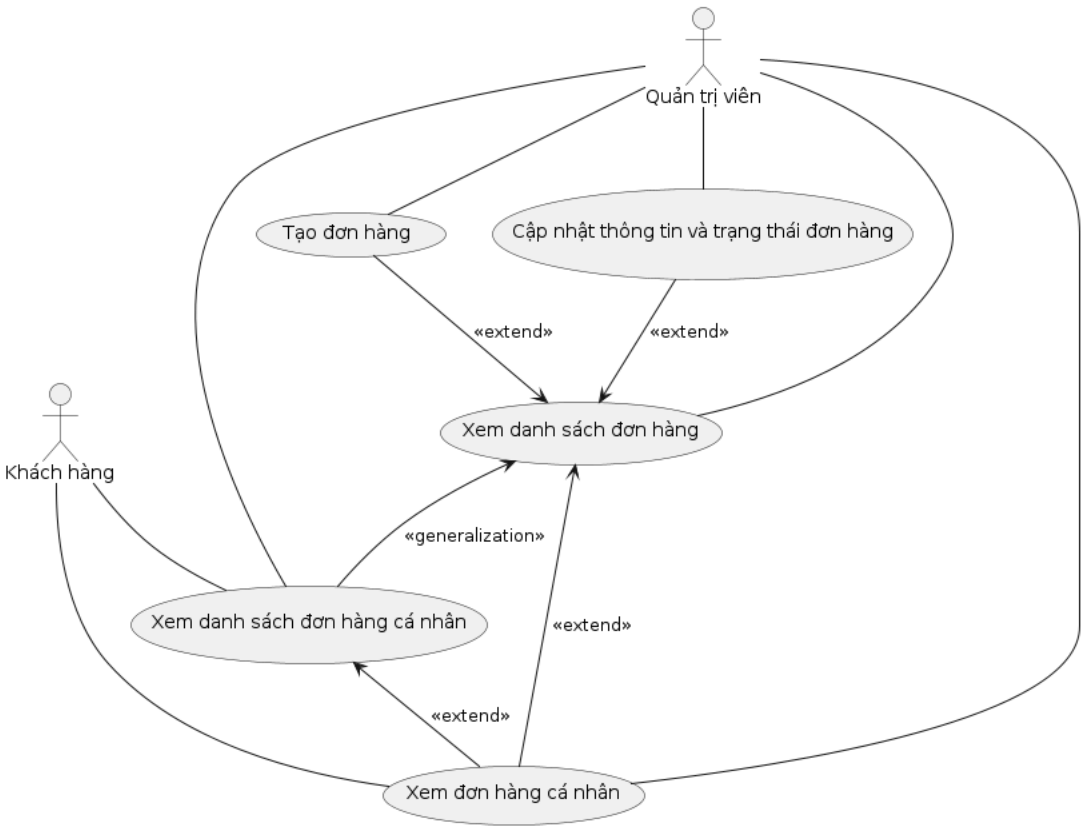
tiết về sản phẩm đã đặt, số lượng, giá cả và thông tin vận chuyển. Ngoài ra, khách hàng cũng có thể hủy bỏ đơn hàng hoặc yêu cầu trả lại sản phẩm trong trường hợp sản phẩm không đáp ứng được yêu cầu của khách hàng.

Danh sách các chức năng con của chức năng quản lý đơn hàng:

- Tạo đơn hàng.
- Xem đơn hàng.
- Cập nhật trạng thái đơn hàng.



Hình 7: Biểu đồ phân rã chức năng quản lý đơn hàng



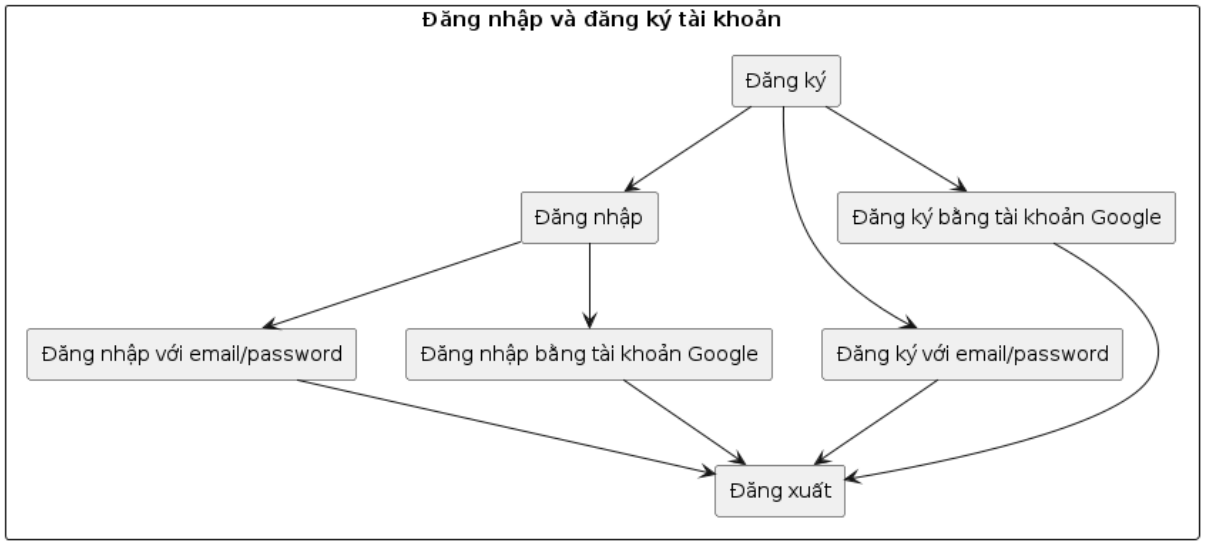
Hình 8: Biểu đồ usecase cho chức năng quản lý đơn hàng

2.1.3. Chức năng đăng ký và đăng nhập tài khoản

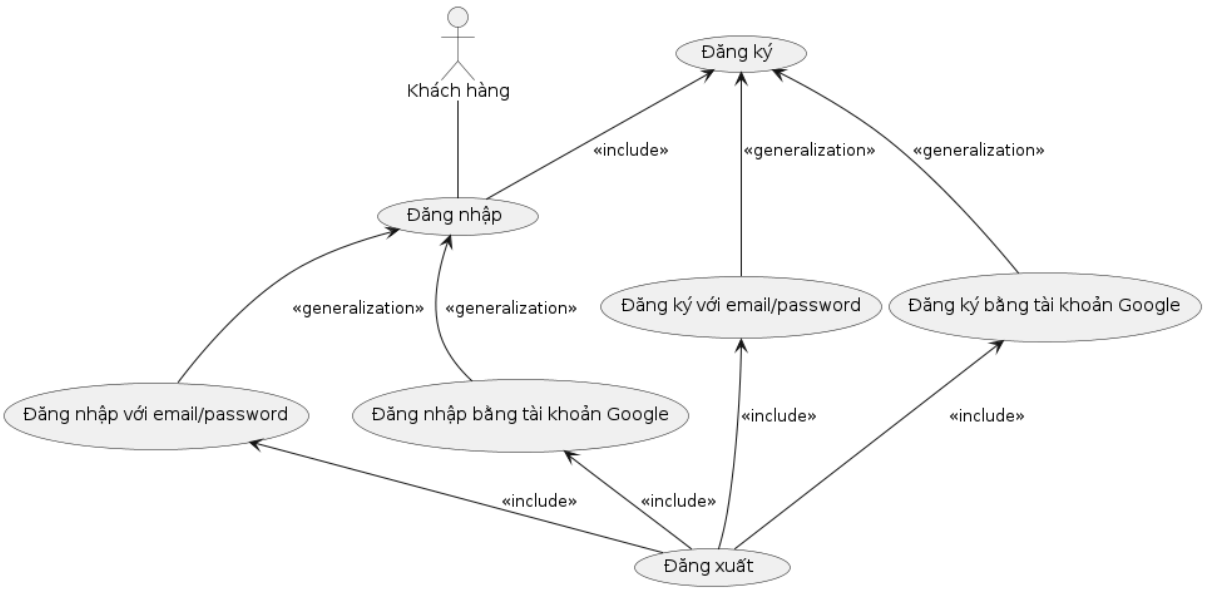
Chức năng đăng ký tài khoản cho phép khách hàng có thể tạo một tài khoản mới trên trang web TMĐT. Người dùng cần cung cấp thông tin cá nhân như tên địa chỉ email, mật khẩu. Thông tin này sẽ được lưu trữ trong hệ thống của trang web để khách hàng có thể đăng nhập lại vào lần sau. Ngoài ra website còn được tích hợp tính năng đăng nhập bằng tài khoản Google sử dụng OAuth 2.0.

Chức năng đăng nhập cung cấp cho khách hàng quyền truy cập vào các tính năng và dịch vụ của trang web TMĐT. Người dùng sẽ cần nhập tên đăng nhập và mật khẩu của mình để đăng nhập thành công. Sau khi đăng nhập thành công, khách hàng có thể thực hiện các hoạt động như xem lịch sử giao dịch, sửa đổi thông tin cá nhân, quản lý giỏ hàng và thanh toán đơn hàng.

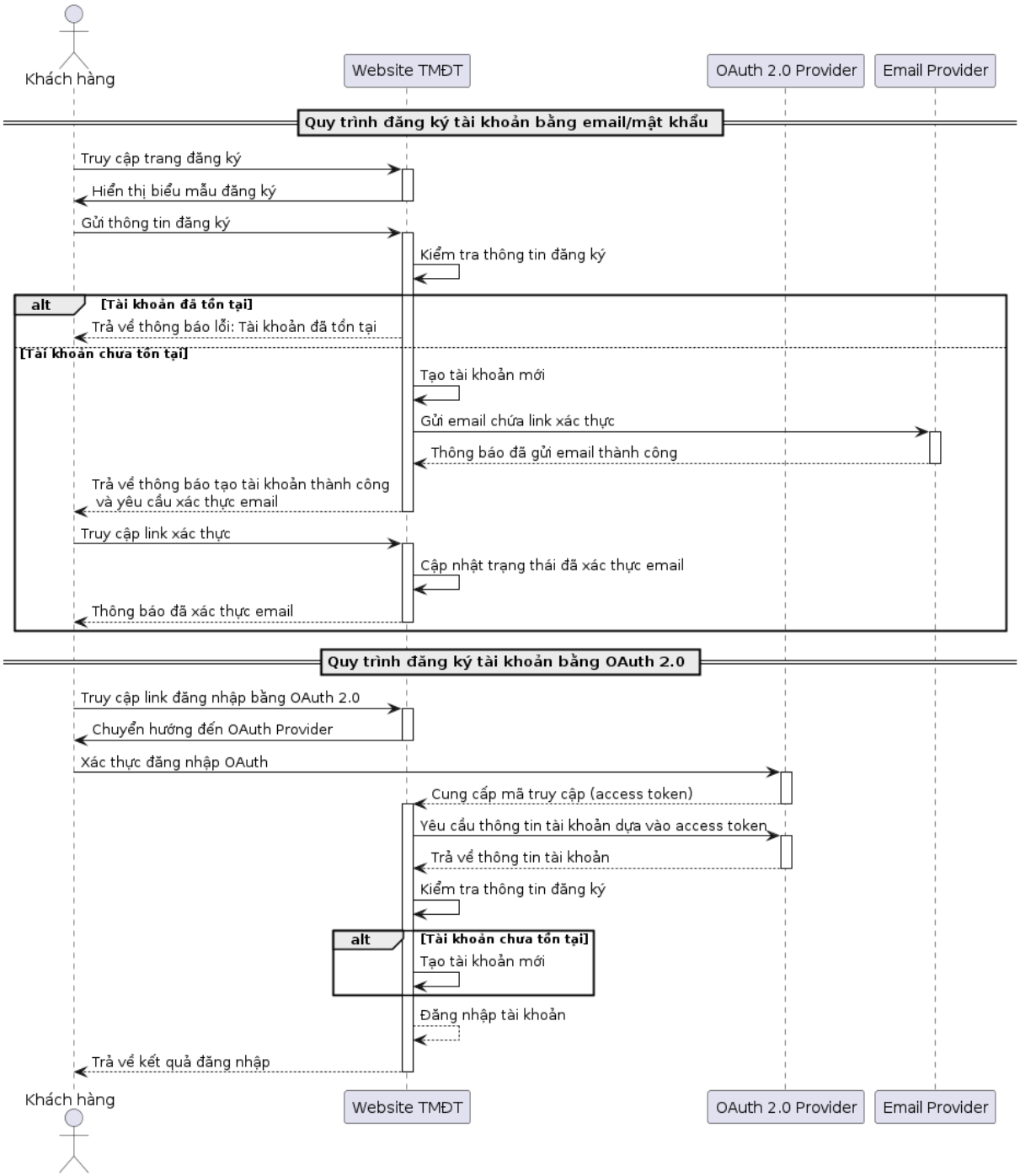
Ngoài ra, việc có chức năng đăng nhập và đăng ký tài khoản còn giúp cho trang web TMĐT có thể thu thập thông tin về khách hàng để có thể cung cấp các dịch vụ tốt hơn và phù hợp với nhu cầu của từng khách hàng.



Hình 9: Biểu đồ phân rã chức năng của chức năng đăng ký và đăng nhập



Hình 10: Biểu đồ usecase cho chức năng đăng nhập và đăng ký tài khoản



Hình 11: Biểu đồ sequence thể hiện luồng xử lý của tính năng đăng ký

Một số cơ chế an toàn trong chức năng này:

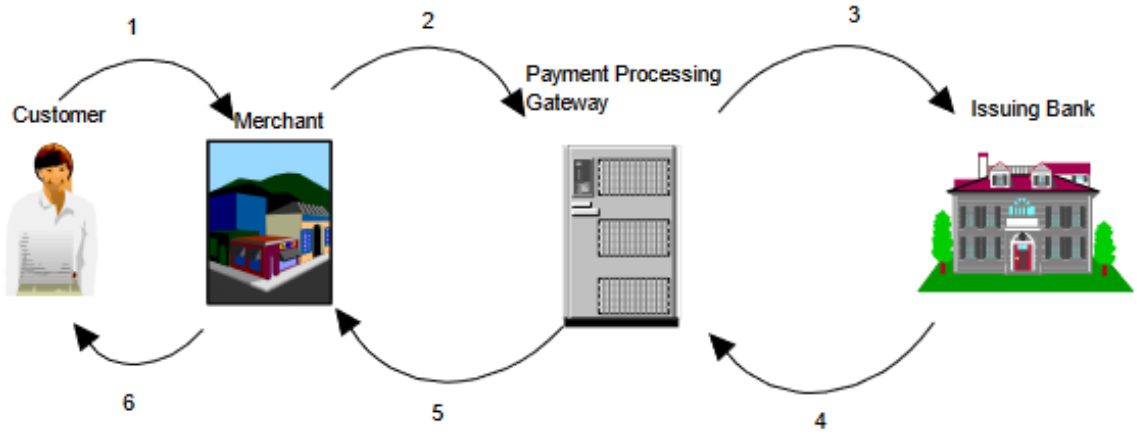
- Mã hóa mật khẩu: Mật khẩu của người dùng nên được mã hóa trước khi lưu trữ trong cơ sở dữ liệu. Một cách thông thường là sử dụng thuật toán băm (hashing) như bcrypt để mã hóa mật khẩu, đảm bảo rằng mật khẩu không thể được khôi phục ngược lại từ các giá trị được lưu trữ.

• Bảo mật giao tiếp: Đảm bảo rằng thông tin đăng nhập và thông tin cá nhân của người dùng được bảo vệ trong quá trình truyền tải. Sử dụng kết nối an toàn qua HTTPS (SSL/TLS) để mã hóa dữ liệu giao tiếp giữa người dùng và hệ thống.

• Kiểm tra mật khẩu mạnh: Yêu cầu người dùng sử dụng mật khẩu mạnh, bao gồm sự kết hợp của ký tự chữ hoa, chữ thường, chữ số và ký tự đặc biệt.

• Bảo vệ thông tin cá nhân: Đảm bảo rằng thông tin cá nhân của người dùng được bảo vệ và không được tiết lộ cho bất kỳ bên thứ ba nào. Áp dụng các biện pháp bảo mật phù hợp như mã hóa dữ liệu, kiểm soát quyền truy cập và chứng thực đúng người dùng để bảo vệ thông tin cá nhân.

2.1.4. Chức năng giỏ hàng và thanh toán



Hình 12: Giao dịch thanh toán trên internet

Đầu tiên, xác định các yêu cầu kinh doanh liên quan đến thanh toán điện tử của website. Website TMĐT trong báo cáo này hỗ trợ các phương thức thanh toán thông qua ví điện tử và thanh toán khi nhận hàng (COD). Đơn vị tiền tệ sử dụng Việt Nam Đồng.

Lựa chọn cổng thanh toán cho ví điện tử: Dựa trên yêu cầu kinh doanh, chọn cổng thanh toán phù hợp để tích hợp vào website. Các cổng thanh toán phổ biến bao gồm PayPal, Stripe,… Sau quá trình khảo sát tác giả quyết định chọn Zalopay và Paypal là hai cổng thanh toán ví điện tử của website.

Tiếp theo cần thiết kế cơ sở dữ liệu để lưu trữ thông tin liên quan đến thanh toán. Ở phần thiết kế danh sách các thực thể chính của website tác giả đã thiết kế thực thể "Payment Method" đại diện cho thông tin các cổng thanh toán được lưu trữ trong website. Từ đó có thể triển khai xây dựng cơ sở dữ liệu ở Chương 3.

Tiếp theo cần thiết kế giao diện người dùng để hiển thị thông tin liên quan đến thanh toán, bao gồm các biểu mẫu nhập thông tin thanh toán, hiển thị đơn hàng, tóm tắt thanh toán và xác nhận thanh toán.

Xử lý và xác thực thanh toán: Khi người dùng hoàn tất thông tin thanh toán, thực hiện xử lý và xác thực thanh toán. Điều này bao gồm gửi thông tin thanh toán đến cổng thanh toán, kiểm tra tính hợp lệ và xác nhận thanh toán từ cổng thanh toán. Nếu thanh toán thành công, cập nhật trạng thái đơn hàng và lưu trữ thông tin thanh toán. Ngoài ra cần xây dựng cơ chế để xử lý các lỗi thanh toán và quản lý giao dịch không thành công bao gồm xử lý các thông báo lỗi từ cổng thanh toán, gửi thông báo cho người dùng và cập nhật trạng thái đơn hàng tương ứng.

Cuối cùng cần kiểm tra toàn bộ quy trình thanh toán để đảm bảo tính chính xác và hoạt động một cách trơn tru trên website. Sau đó, triển khai mô hình xử lý thanh toán vào website.

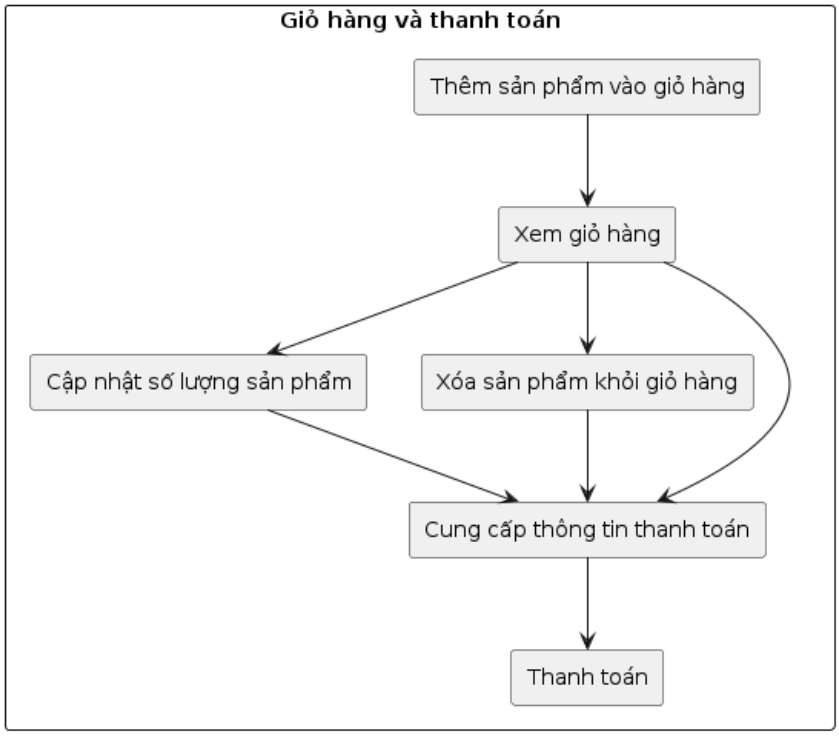
Chức năng giỏ hàng cho phép người dùng lưu trữ các sản phẩm mà họ muốn mua vào trong giỏ hàng. Người dùng có thể thêm hoặc xóa bất kỳ sản phẩm nào từ giỏ hàng của mình và có thể xem toàn bộ giỏ hàng của mình trước khi hoàn tất đơn hàng.

Sau khi đã chọn các sản phẩm mua, khách hàng cần thực hiện thanh toán để hoàn tất đơn hàng. Chức năng thanh toán cung cấp cho khách hàng các phương thức thanh toán khác nhau để lựa chọn, bao gồm thanh toán qua ví Zalopay, Paypal, thanh toán COD (thanh toán khi nhận hàng).

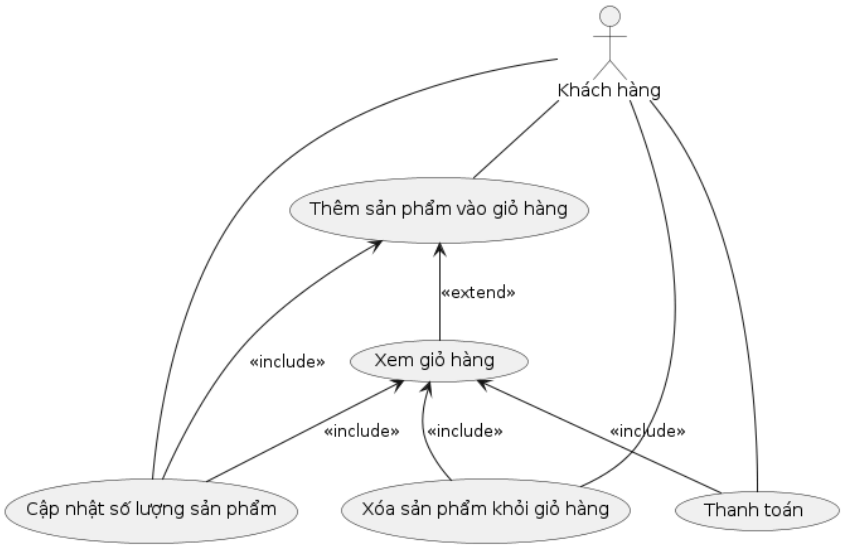
Ngoài ra, trang web TMĐT cũng cần đảm bảo rằng các thông tin thanh toán của khách hàng được bảo mật và an toàn. Vì vậy, trang web TMĐT cần sử dụng các công nghệ bảo mật như SSL (Secure Sockets Layer) để mã hóa thông tin thanh toán và tránh các vấn đề bảo mật như lừa đảo hoặc giả mạo thông tin.

Danh sách các chức năng con của chức năng giỏ hàng và thanh toán:

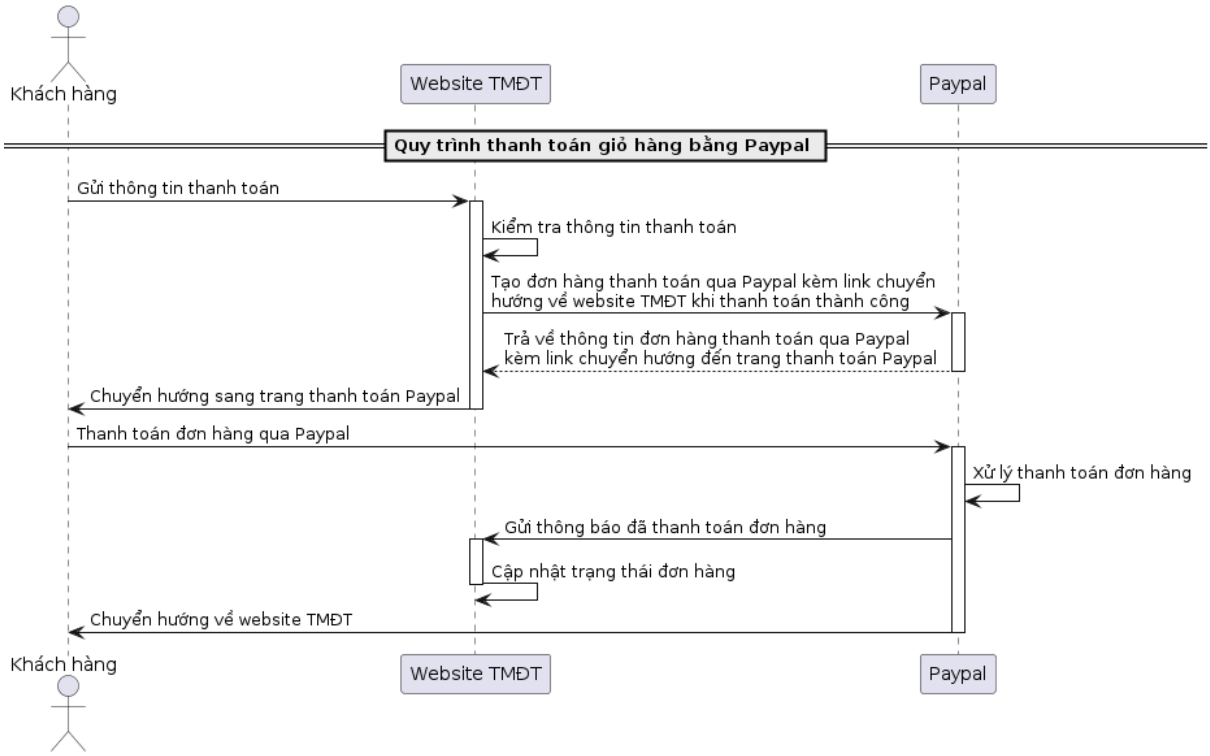
- Thêm sản phẩm vào giỏ hàng.
- Cập nhật số lượng sản phẩm.
- Xóa sản phẩm ra khỏi giỏ hàng.
- Thanh toán.



Hình 13: Biểu đồ phân rã chức năng giỏ hàng và thanh toán



Hình 14: Biểu đồ usecase cho chức năng giỏ hàng và thanh toán



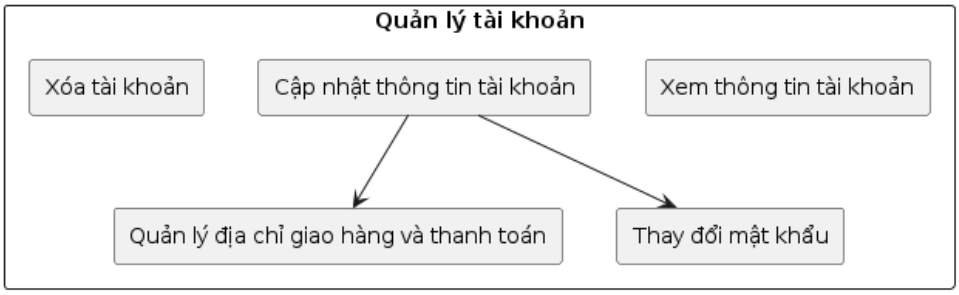
Hình 15: Biểu đồ sequence thể hiện luồng thực hiện tính năng thanh toán với cổng thanh toán Paypal

2.1.5. Chức năng quản lý tài khoản

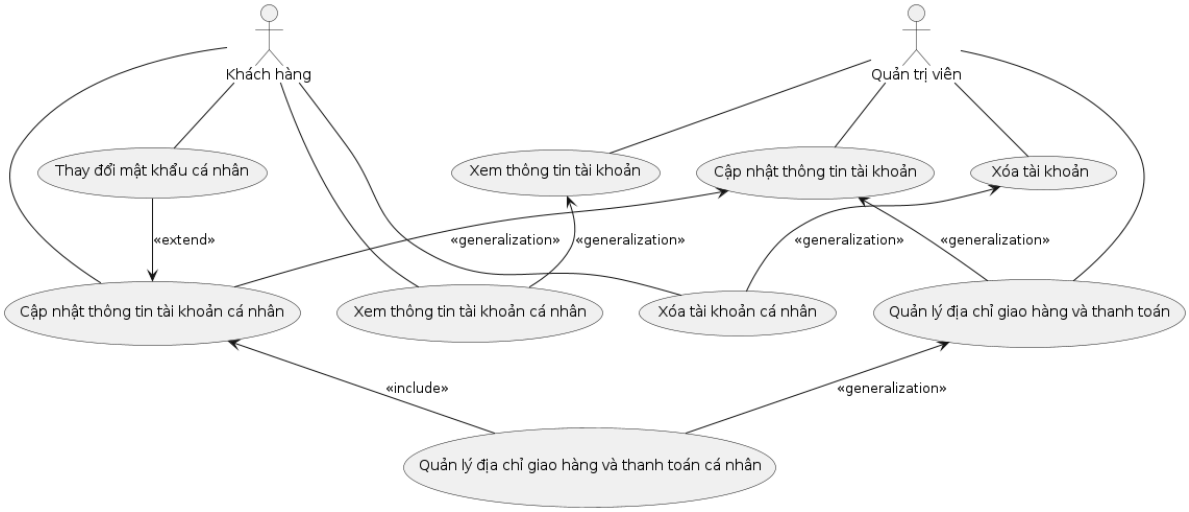
Chức năng quản lý tài khoản cho phép quản trị viên quản lý danh sách khách hàng trên website TMĐT và cho phép khách hàng cập nhật xem và sửa đổi thông tin cá nhân của mình, bao gồm tên, địa chỉ, số điện thoại và địa chỉ email. Khách hàng cũng có thể thay đổi mật khẩu để bảo mật tài khoản của mình.

Danh sách các chức năng con của chức năng quản lý thông tin tài khoản:

- Xem danh sách tài khoản.
- Xem thông tin tài khoản.
- Cập nhật thông tin tài khoản.
- Xóa tài khoản.



Hình 16: Biểu đồ phân rã tinh năng quản lý tài khoản



Hình 17: Biểu đồ usecase của chức năng quản lý tài khoản

2.2. Thiết kế cơ sở dữ liệu

2.2.1. Lựa chọn hệ quản trị cơ sở dữ liệu

Lựa chọn hệ quản trị cơ sở dữ liệu (Database Management System - DBMS) trong thiết kế website thương mại điện tử là một yếu tố quan trọng và có ảnh hưởng đáng kể đến hiệu suất, bảo mật và khả năng mở rộng của hệ thống.

Phân loại và đánh giá tổng quát của các loại cơ sở dữ liệu phổ biến:

- Relational Database:
    - Đặc điểm: Cơ sở dữ liệu quan hệ sử dụng bảng và mối quan hệ để tổ chức dữ liệu thành các thực thể và quan hệ giữa chúng. Structured

Query Language (SQL) là ngôn ngữ phổ biến để truy vấn và quản lý cơ sở dữ liệu quan hệ.

- Ưu điểm: Dễ hiểu, dễ sử dụng, hỗ trợ các hoạt động truy vấn phức tạp, đảm bảo tính nhất quán và toàn vẹn dữ liệu.
- Nhược điểm: Cần sử dụng quá nhiều liên kết giữa các bảng trong trường hợp dữ liệu phức tạp, có thể làm chậm hiệu suất truy vấn.

- Non-Relational Database:
  - Đặc điểm: Các cơ sở dữ liệu không quan hệ, hay còn gọi là NoSQL, có cấu trúc linh hoạt hơn và không sử dụng mô hình quan hệ. Các loại cơ sở dữ liệu NoSQL bao gồm: cơ sở dữ liệu cột, cơ sở dữ liệu tài liệu, cơ sở dữ liệu đồ thị và cơ sở dữ liệu key-value.
  - Ưu điểm: Khả năng mở rộng tốt, hỗ trợ truy vấn nhanh, linh hoạt và có thể xử lý dữ liệu phi cấu trúc.
  - Nhược điểm: Không đảm bảo tính nhất quán dữ liệu như cơ sở dữ liệu quan hệ, hạn chế trong việc truy vấn phức tạp.

- Graph Database:
  - Đặc điểm: Cơ sở dữ liệu đồ thị được sử dụng để lưu trữ dữ liệu có mối quan hệ phức tạp. Nó sử dụng các nút, cạnh và thuộc tính để biểu diễn dữ liệu và quan hệ giữa chúng.
  - Ưu điểm: Hiệu suất cao trong việc truy vấn và phân tích các mối quan hệ dữ liệu phức tạp.
  - Nhược điểm: Thường không phù hợp cho các dự án có dữ liệu đơn giản hoặc ít quan hệ.

Dựa vào các phân tích và đánh giá được thực hiện trong Chương 1, kết hợp tìm hiểu trong quá trình xây dựng yêu cầu nghiệp vụ và chức năng của website, tác giả đánh giá rằng mô hình dữ liệu của website được liên kết với nhau một cách chặt chẽ và mạch lạc. Do đó, tác giả đã quyết định chọn SQL làm hệ quản trị cơ sở dữ liệu cho website, bởi SQL mang đến những ưu điểm quan trọng và phù hợp với yêu cầu của dữ liệu trong dự án.

Bằng việc sử dụng SQL, website thương mại điện tử sẽ có khả năng xử lý dữ liệu một cách hiệu quả, đáng tin cậy và linh hoạt. SQL cung cấp một ngôn ngữ truy vấn mạnh mẽ cho phép thực hiện các truy vấn phức tạp và tùy chỉnh theo nhu cầu cụ thể của dự án.

2.2.2. Thiết kế mô hình dữ liệu

Trong quá trình phân tích thiết kế website thương mại điện tử, thiết kế mô hình dữ liệu đóng vai trò quan trọng để xác định cấu trúc và quan hệ giữa các bảng dữ liệu. Việc thiết kế mô hình dữ liệu cẩn thận và hợp lý là yếu tố quyết định thành công của hệ thống cơ sở dữ liệu, ảnh hưởng đến hiệu suất, tính nhất quán và quản lý dữ liệu. Dưới đây là phân tích và đánh giá chi tiết về thiết kế mô hình dữ liệu trong phân tích thiết kế website thương mại điện tử:

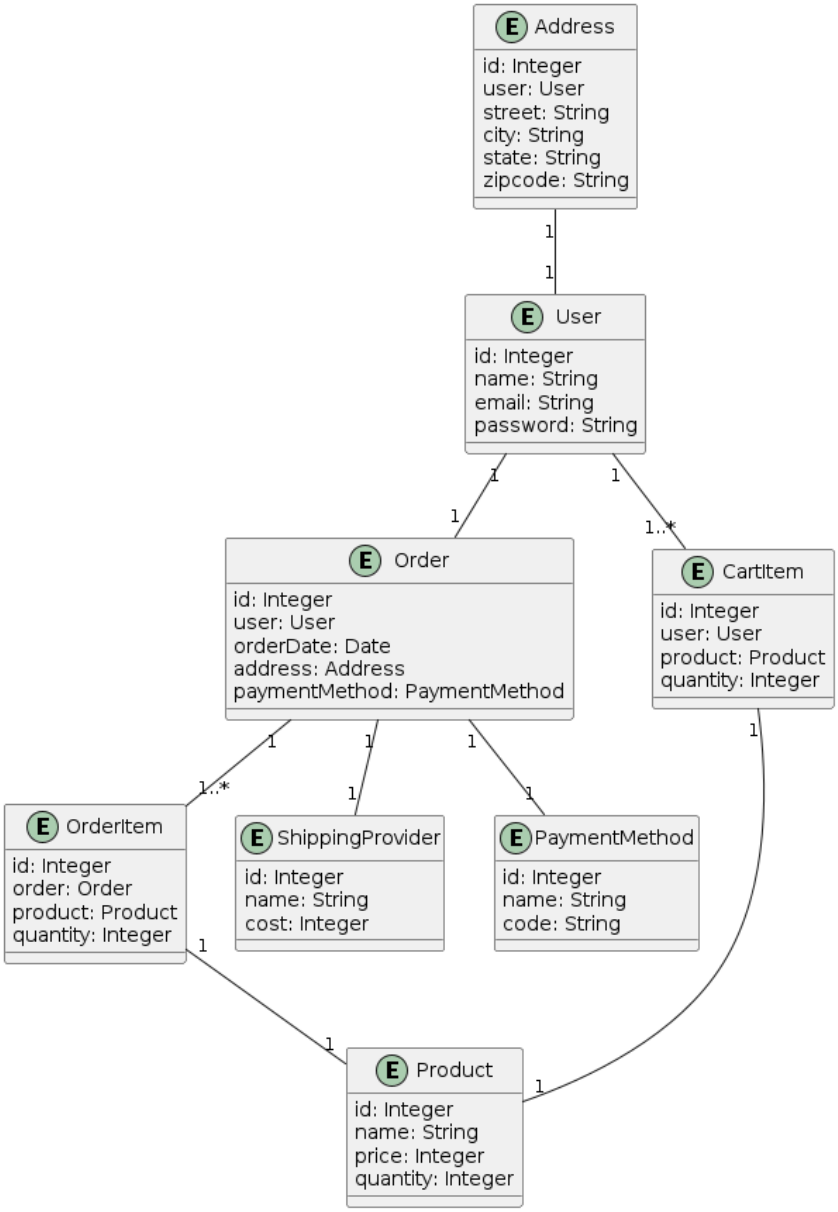
- Xác định yêu cầu và mục tiêu: Trước khi thiết kế mô hình dữ liệu, cần xác định rõ yêu cầu và mục tiêu của website thương mại điện tử. Điều này bao gồm việc hiểu rõ các chức năng, quy trình kinh doanh, quyền hạn và yêu cầu về dữ liệu của hệ thống.
- Xác định các thực thể (entities): Xác định các thực thể chính trong hệ thống, như sản phẩm, người dùng, đơn hàng, danh mục sản phẩm, v.v. Mỗi thực thể đại diện cho một tập hợp các đối tượng có liên quan trong thế giới thực.
- Xác định các thuộc tính (attributes) của thực thể: Xác định các thuộc tính cần thiết để mô tả và lưu trữ thông tin về các thực thể. Ví dụ, thuộc tính của thực thể "sản phẩm" có thể bao gồm tên, mô tả, giá, hình ảnh, số lượng, v.v.

Từ những đánh giá trên tác giả đã xây dựng danh sách các thực thể chính trong một website TMĐT như sau:

- User: Đại diện cho người dùng của website, bao gồm thông tin như tên, email, mật khẩu, và các thông tin liên quan khác.
- Product: Lưu trữ thông tin về các sản phẩm mà website đang bán, bao gồm tên, mô tả, giá, số lượng, mã SKU, và trạng thái có sẵn hay không.

- Order: Đại diện cho đơn hàng được tạo bởi người dùng, bao gồm thông tin như người đặt hàng, phương thức thanh toán, địa chỉ giao hàng, tổng giá trị đơn hàng, ghi chú và trạng thái đơn hàng.
- Order Item: Lưu trữ thông tin về các sản phẩm được đặt trong mỗi đơn hàng, bao gồm sản phẩm, số lượng và giá.
- Cart Item: Lưu trữ thông tin về các sản phẩm trong giỏ hàng của người dùng, bao gồm sản phẩm, số lượng và người dùng tương ứng.
- Shipping Provider: Đại diện cho các nhà cung cấp dịch vụ giao hàng mà website hỗ trợ.
- Payment Method: Đại diện cho các phương thức thanh toán mà người dùng có thể sử dụng khi đặt hàng.
- Address: Lưu trữ thông tin về địa chỉ của người dùng, bao gồm quốc gia, thành phố, tiểu bang/tỉnh, đường phố, và mã bưu điện.

Sau khi đã xác định các thực thể chính trong website, dựa vào đó tác giả đã thiết kế mô hình ER cho cơ sở dữ liệu để triển khai trong Chương 3:



Hình 18: Biểu đồ ER của CSDL website TMĐT

2.2.3. Đảm bảo tính nhất quán và an toàn cho cơ sở dữ liệu

Đảm bảo tính nhất quán và an toàn cho cơ sở dữ liệu là một yếu tố quan trọng trong phân tích và thiết kế website thương mại điện tử. Điều này đảm bảo rằng dữ liệu được lưu trữ và truy cập một cách đáng tin cậy, đồng thời đảm bảo tính toàn vẹn và bảo mật của thông tin. Dưới đây là phân tích và đánh giá chi tiết về đảm bảo tính nhất quán và an toàn cho cơ sở dữ liệu trong phân tích thiết kế website thương mại điện tử:

- Tính nhất quán của cơ sở dữ liệu.
- Tính toàn vẹn dữ liệu.
- Bảo mật dữ liệu.
- Sao lưu và phục hồi dữ liệu.

2.3. Phân tích thiết kế kiến trúc hệ thống

2.3.1. Xác định các thành phần hệ thống

Từ những phân tích trên kết hợp với việc ứng dụng mô hình MVC vào website TMĐT, ta có các thành phần chính như sau:

- Model: Đại diện cho logic xử lý dữ liệu trong ứng dụng. Model thực hiện truy vấn, thêm, sửa đổi và xóa dữ liệu từ cơ sở dữ liệu. Nó quản lý lưu trữ và truy xuất dữ liệu thông qua truy vấn SQL hoặc Object Relational Mapping (ORM).
- View: Đại diện cho giao diện người dùng của ứng dụng. View hiển thị dữ liệu cho người dùng và tương tác với họ. Nó chứa mã HTML, CSS và JavaScript để tạo giao diện người dùng.
- Controller: Xử lý yêu cầu từ người dùng và tương tác với Model và View. Controller nhận yêu cầu qua Routes và gọi phương thức tương ứng để xử lý. Nó thực hiện lấy dữ liệu từ Model, xử lý logic và chuyển dữ liệu đến View.
- Routes: Xác định các đường dẫn URL và kết nối chúng với phương thức trong Controller. Routes định nghĩa các điểm cuối (endpoints) của ứng dụng và quyết định điều hướng yêu cầu từ người dùng đến Controller.
- Middleware: Là thành phần trung gian giữa yêu cầu và phản hồi của ứng dụng. Middleware cho phép xử lý trước và sau khi yêu cầu đi qua Controller. Nó kiểm soát quyền truy cập, xác thực người dùng, thực hiện xử lý logic và bổ sung thông tin vào yêu cầu.

Thiết kế cấu trúc thư mục của website:

- Thư mục `app` chứa các thành phần chính của ứng dụng như Controllers, Models, và Services.
- Thư mục `config` chứa các tệp cấu hình của ứng dụng.

- Thư mục `database` chứa các file migration và seeders.
- Thư mục `public` chứa các file tĩnh như hình ảnh, CSS, và JavaScript.
- Thư mục `resources` chứa các file nguồn của ứng dụng như views, assets, và ngôn ngữ.
- Thư mục `routes` chứa các file định tuyến của ứng dụng.

2.3.2. Các cơ chế an toàn

Cần xác định rõ các yêu cầu bảo mật của hệ thống. Bao gồm việc xác định các nguyên tắc bảo mật cần tuân thủ, các loại dữ liệu nhạy cảm cần được bảo vệ, và các rủi ro bảo mật tiềm ẩn. Xác định các yêu cầu bảo mật sẽ giúp xác định các khía cạnh cần thiết để bảo vệ thông tin của người dùng.

Tiếp theo, cần thiết kế kiến trúc hệ thống hướng bảo mật. Bao gồm việc sử dụng các phương pháp mã hóa mạnh mẽ để bảo vệ thông tin, xác thực và ủy quyền để kiểm soát quyền truy cập, và các lớp bảo vệ phòng ngừa tấn công từ bên ngoài.

Một phần quan trọng trong việc đảm bảo tính bảo mật là quản lý danh sách truy cập và phân quyền. Cần xác định các vai trò người dùng và quyền hạn tương ứng để kiểm soát quyền truy cập vào hệ thống và dữ liệu. Quản lý danh sách truy cập và phân quyền giúp đảm bảo rằng chỉ những người được ủy quyền mới có thể truy cập và thao tác với thông tin nhạy cảm.

Cần thực hiện kiểm tra và xác minh bảo mật thường xuyên để đảm bảo tính bảo mật của hệ thống. Kiểm tra bảo mật bao gồm việc kiểm tra các lỗ hổng bảo mật và điểm yếu trong hệ thống, trong khi xác minh bảo mật đảm bảo rằng các biện pháp bảo mật đã được triển khai và hoạt động hiệu quả. Kiểm tra và xác minh bảo mật định kỳ giúp phát hiện và khắc phục các lỗ hổng bảo mật sớm trước khi chúng có thể bị tấn công.

Cuối cùng, cần đảm bảo rằng nhân viên được đào tạo và có nhận thức về bảo mật. Đào tạo nhân viên về các quy trình bảo mật và các biện pháp bảo mật sẽ giúp nâng cao ý thức về bảo mật và đảm bảo rằng mọi người đều thực hiện các biện pháp bảo mật một cách đúng đắn.

Một số phương pháp đảm bảo an toàn cho ứng dụng cho website TMĐT:

• Sử dụng giao thức an toàn Secure Socket Layer (SSL): SSL/TLS tạo ra một kênh truyền thông an toàn giữa máy tính của người dùng và máy chủ của website, đảm bảo rằng dữ liệu không bị đánh cắp, thay đổi hoặc giả mạo trong quá trình truyền. Khi một trang web sử dụng SSL/TLS, địa chỉ URL sẽ bắt đầu bằng `https://` thay vì `http://`, và thông thường trình duyệt sẽ hiển thị một biểu tượng khóa hoặc dấu chấm than xanh lá để chỉ ra rằng kết nối là an toàn. Sử dụng giao thức HTTPS là một cách triển khai SSL. Một số nhà cung cấp chứng chỉ SSL miễn phí để sử dụng giao thức HTTPS cho website như Cloudflare và Let's Encrypt. Để cài đặt chứng chỉ cho website ta cần vào trang quản trị tên miền trỏ nameserver về nameserver của nhà cung cấp (đối với Cloudflare) hoặc cài đặt các khóa chứng chỉ của nhà cung cấp lên VPS mà tên miền đang trỏ về (đối với Let's Encrypt).

• Mã hóa và xác thực với Tokenization: Thông tin thanh toán nhạy cảm của người sử dụng được thay thế bằng một tập hợp các ký tự được gọi là token và các token này sẽ không ảnh hưởng đến tính an toàn trong các giao dịch trực tuyến và di động. Các máy khách sẽ thực hiện truyền mã token, thay vì dữ liệu thông tin gốc quan trọng, điều này khiến dữ liệu sẽ không thể bị đánh cắp hoặc không có giá trị đối với kẻ tấn công khi đánh cắp được. Một trong số cách triển khai đó là xác thực người dùng sử dụng JSON Web Token (JWT) là một phương thức xác thực dựa trên mã thông báo (token) được tạo và ký bởi máy chủ. JWT hỗ trợ nhiều thuật toán mã hóa phổ biến như HS256 (HMAC SHA-256), RS256 (RSA SHA-256), và nhiều thuật toán khác dựa trên mã hóa đối xứng và bất đối xứng.

• Thiết kế theo tiêu chuẩn Payment Card Industry Data Security Standard (PCI DSS): Để đạt được tuân thủ PCI DSS, website phải thực hiện theo một loạt các yêu cầu khắt khe, chẳng hạn như thực hiện bảo mật hệ thống và mạng để bảo vệ dữ liệu thẻ tín dụng, bảo vệ các thông tin xác thực của khách hàng bằng cách sử dụng các giải pháp mã hóa và thực hiện quản lý quy trình và chính sách bảo mật, đảm bảo rằng nhân viên được đào tạo và thực hiện theo tiêu chuẩn an ninh thông tin.

• Xác thực với giao thức Open Authorization: Open Authorization (OAuth) là một giao thức xác thực và ủy quyền được sử dụng để cho phép người dùng

cấp quyền truy cập tài khoản của mình cho các ứng dụng, dịch vụ và trang web khác. OAuth cho phép người dùng chia sẻ thông tin cá nhân và tài khoản của họ mà không cần tiết lộ mật khẩu của mình. Thay vào đó, OAuth sử dụng một mã truy cập để cung cấp quyền truy cập. Khi người dùng cấp quyền truy cập cho ứng dụng, dịch vụ hoặc trang web, mã truy cập sẽ được tạo ra. Mã này sau đó được sử dụng để xác thực yêu cầu truy cập từ ứng dụng, dịch vụ hoặc trang web đó. OAuth 2.0 là phiên bản tiếp theo của giao thức OAuth. OAuth 2.0 giúp cho việc ủy quyền truy cập dễ dàng hơn và cung cấp nhiều cấp độ quyền hơn so với phiên bản trước đó.

- Quản lý phiên làm việc bằng việc triển khai session cho website. Framework Laravel đã hỗ trợ session cho user.
- Chỉ lưu trữ mật khẩu đã mã hóa 1 chiều vào website nhằm giảm khả năng lộ lọt dữ liệu của cơ sơ dữ liệu, có thể sử dụng hàm mã hóa `bcrypt` của Laravel.
- Kiểm tra và sửa lỗi bảo mật: Thực hiện kiểm tra bảo mật định kỳ, bao gồm kiểm tra lỗ hổng, kiểm tra xác thực, kiểm tra cấu hình, và kiểm tra mã nguồn để tìm và sửa các lỗi bảo mật tiềm ẩn.
- Xây dựng các bộ kiểm thử input đầu vào. Ta có thể sử dụng class `Validator` được hỗ trợ bởi framework Laravel hoặc có thể xây dựng các kịch bản kiểm thử input đầu vào. Dưới đây là một số bộ kiểm thử input đầu vào có thể sử dụng trong website:

```
// Thông tin thanh toán đơn hàng
$rules = [
    'name' => 'required',
    'email' => 'required|email',
    'country' => 'required',
    'city' => 'required',
    'state' => 'required',
    'street' => 'required',
    'postalCode' => 'required|postal_code',
    'phone' => 'required|vn_phone_number',
    'paymentMethodId' => 'required',
    'cart' => 'required|array|min:1',
];

// Thông tin đăng ký
$rules = [
```

```
    'name' => 'required|string|max:255',
    'email' => 'required|email|unique:users',
    'password' => 'required|string|min:8|confirmed',
];

// Thông tin cập nhật tài khoản
$rules = [
    'name' => 'required|string|max:255',
    'email' => 'required|email|unique:users,email,'.$id,
];
```

2.4. Giải pháp xây dựng giao diện và trải nghiệm người dùng

2.4.1. Xây dựng giao diện với Tailwind

Tailwind một framework CSS utility-first được lựa chọn vì sự linh hoạt và hiệu quả trong việc thiết kế giao diện. Tailwind CSS đặc biệt hữu ích cho việc xây dựng giao diện web từ đầu, mang lại khả năng tùy chỉnh cao và sự linh hoạt trong việc tạo ra các thành phần giao diện.

Một trong những ưu điểm của Tailwind CSS là phương pháp thiết kế utility-first. Thay vì tạo ra các class CSS đặc biệt cho từng thành phần, Tailwind CSS tập trung vào việc cung cấp các class utilitarian như `w-`, `h-`, `bg-`, `text-`,... cho phép xây dựng giao diện nhanh chóng bằng cách kết hợp các class này lại với nhau. Điều này giúp tiết kiệm thời gian viết CSS từ đầu và cho phép tập trung vào việc xây dựng giao diện một cách hiệu quả.

Ngoài ra, Tailwind CSS cung cấp một design system rất phong phú với rất nhiều thành phần giao diện sẵn có. Các thành phần này bao gồm nút, thẻ, menu, biểu đồ, bảng và nhiều hơn nữa.

Một khía cạnh quan trọng khác của việc sử dụng Tailwind CSS là khả năng áp dụng Responsive Web Design (RWD). Framework cung cấp các class CSS breakpoint như `sm`, `md`, `lg`, `xl` để giúp điều chỉnh giao diện theo kích thước màn hình khác nhau. Bằng cách sử dụng các class này, ta có thể dễ dàng điều chỉnh giao diện để nó hiển thị tốt trên điện thoại di động, máy tính bảng và máy tính để bàn.

Đối với trải nghiệm người dùng, Tailwind CSS cung cấp một giao diện đẹp, dễ nhìn và dễ sử dụng. Các thành phần được thiết kế sao cho tương thích với nguyên tắc thiết kế giao diện hiện đại và hướng tới trải nghiệm người dùng tốt nhất. Ta có thể tùy chỉnh giao diện để phù hợp với thương hiệu và mục tiêu của mình, tạo nên một trải nghiệm độc đáo và chuyên nghiệp cho người dùng.

2.4.2. Tối ưu hóa tốc độ load trang

Tối ưu hóa tốc độ load trang là một trong những yếu tố quan trọng để cải thiện trải nghiệm người dùng và tăng tương tác trên website TMĐT. Dưới đây là một số cách để tối ưu hóa tốc độ load trang:

- Tối ưu hóa hình ảnh: Sử dụng các công cụ tối ưu hóa hình ảnh để giảm dung lượng của các hình ảnh trên trang web, đồng thời áp dụng kỹ thuật lazy loading để chỉ tải hình ảnh khi cần thiết.
- Sử dụng cache: Sử dụng bộ nhớ cache để giảm thời gian tải lại trang web và cải thiện trải nghiệm người dùng.
- Giảm số lượng yêu cầu HTTP: Giảm số lượng yêu cầu HTTP bằng cách sử dụng các kỹ thuật như gộp file CSS và JavaScript hoặc sử dụng các CDN (Content Delivery Network) để phân phối tài nguyên trên nhiều máy chủ.
- Chọn hosting tốt: Lựa chọn một nhà cung cấp hosting tốt có thể giúp tăng tốc độ tải trang web.
- Tối ưu hóa mã nguồn: Sử dụng các phương pháp tối ưu hóa mã nguồn, chẳng hạn như sử dụng minifier để giảm kích thước của mã HTML, CSS và JavaScript.
- Sử dụng các công cụ đo lường hiệu suất: Sử dụng các công cụ đo lường hiệu suất như Google PageSpeed Insights để theo dõi và đánh giá tốc độ tải trang web.

Tuy nhiên, việc tối ưu hóa tốc độ load trang là một quá trình liên tục và cần được thực hiện thường xuyên để đạt được hiệu quả tối đa.

2.5. Kết chương

Trong chương 2 tác giả đã trình bày các bước xây dựng mô hình cơ sở dữ liệu, thiết kế kiến trúc dự án và các quy trình tích hợp cổng thanh toán cũng như quy trình cụ thể để đảm bảo an toàn dữ liệu cho website TMĐT. Việc phân tích và đánh giá trước khi bắt đầu xây dựng sản phẩm giúp cho việc xây dựng trở nên an toàn và hiệu quả, đạt được lợi ích lớn nhất. Từ những phân tích và đánh giá đó sẽ áp dụng vào việc xây dựng website sẽ được trình bày trong Chương 3.

CHƯƠNG 3. XÂY DỰNG SẢN PHẨM

3.1. Chuẩn bị môi trường phát triển

Bảng 1: Môi trường phát triển website TMĐT

OS	Linux
Web hosting control panel	cPanel
Webserver	Apache
Version control system	Git, Github
Web framework	Laravel
Database	MySQL
IDE	Visual Studio Code

Việc chuẩn bị môi trường phát triển là rất quan trọng trong quá trình phát triển của một dự án web. Có nhiều yếu tố ảnh hưởng đến việc lựa chọn công nghệ: chi phí, chất lượng nhân lực, tính mở rộng, sự phổ biến và hỗ trợ từ cộng đồng của công nghệ đó,… Việc lựa chọn công nghệ phù hợp với dự án giúp cho việc phát triển dự án nhanh chóng, hiệu quả và ít rủi ro hơn.

Sau nhiều lần tìm hiểu và cân nhắc, tác giả quyết định sử dụng Laravel là một web framework hỗ trợ rất mạnh mẽ trong việc xây dựng website. Laravel được sử dụng rất phổ biến trong cộng đồng lập trình viên web vì sự mạnh mẽ cũng như hỗ trợ và cập nhật rất tốt từ tác giả và cộng đồng.

Về IDE thì lựa chọn phổ biến nhất cho lập trình viên web đó là VS Code do đây là IDE được sử dụng phổ biến và được hỗ trợ rất tốt từ cộng đồng với khả năng tùy biến cao và nhiều plugin kèm theo. Ngoài ra còn do cá nhân tác giả đã có nhiều kinh nghiệm sử dụng VS Code. Đây là lựa chọn thuộc về chất lượng nhân lực.

Về phần hạ tầng tác giả chủ trương sử dụng web hosting để tiết kiệm chi phí và dễ dàng bảo trì, do đó đi kèm theo là sử dụng hệ điều hành Linux, cPanel

control panel, MySQL database và Apache web server do đây là 4 service kèm theo phổ biến của shared web hosting và cũng phù hợp với nhu cầu dự án.

3.2. Xây dựng cơ sở dữ liệu

Mã SQL trên định nghĩa các bảng trong cơ sở dữ liệu MySQL. Dưới đây là danh sách các bảng và chức năng của chúng:

Bảng `addresses`:

- Chức năng: Lưu trữ thông tin địa chỉ.
- Các cột:
    - `id`: Khóa chính, số nguyên không dấu, tự động tăng.
    - `country`: `varchar(255)`, không được để trống.
    - `city`: `varchar(255)`, không được để trống.
    - `state`: `varchar(255)`, không được để trống.
    - `street`: `varchar(255)`, không được để trống.
    - `zip_code`: `varchar(255)`, không được để trống.

Bảng `cart_items`:

- Chức năng: Lưu trữ thông tin về các mục trong giỏ hàng.
- Các cột:
    - `id`: Khóa chính, số nguyên không dấu, tự động tăng.
    - `product_id`: Khóa ngoại đến bảng `products` (id).
    - `user_id`: Khóa ngoại đến bảng `users` (id).
    - `quantity`: Số nguyên không dấu, không được để trống.
    - `created_at`: Thời điểm tạo, kiểu timestamp, không được để trống.
    - `updated_at`: Thời điểm cập nhật, kiểu timestamp, không được để trống.
    - `deleted_at`: Thời điểm xóa, kiểu timestamp, có thể là null.

Bảng `fulfilled_orders`:

- Chức năng: Lưu trữ thông tin về các đơn hàng đã hoàn tất.

- Các cột:
    - `id`: Khóa chính, số nguyên không dấu, tự động tăng.
    - `phone`: `varchar(255)`, không được để trống.
    - `email`: `varchar(255)`, không được để trống.
    - `total`: Số thực không dấu, không được để trống.
    - `status`: Số nguyên không dấu, không được để trống.
    - `created_at`: Thời điểm tạo, kiểu timestamp, có thể là null.
    - `updated_at`: Thời điểm cập nhật, kiểu timestamp, có thể là null.
    - `shipping_provider_id`: Khóa ngoại đến bảng "shipping_providers" (id).
    - `tracking_id`: `varchar(255)`, không được để trống.
    - `note`: `varchar(255)`, có thể là null.
    - `shipping_cost`: Số thực không dấu, không được để trống.
    - `address_id`: Khóa ngoại đến bảng "addresses" (id).

Bảng `orders`:

- Chức năng: Lưu trữ thông tin về các đơn hàng.
- Các cột:
    - `id`: Khóa chính, số nguyên không dấu, tự động tăng.
    - `payment_method_id`: Khóa ngoại đến bảng `payment_methods` (id).
    - `name`: `varchar(255)`, không được để trống.
    - `email`: `varchar(255)`, không được để trống.
    - `phone`: `varchar(255)`, không được để trống.
    - `total`: Số thực không dấu, không được để trống.
    - `note`: Kiểu văn bản, có thể là null.
    - `status`: ENUM('unpaid', 'processing', 'paid', 'cancelled', 'cod'), không được để trống.
    - `user_id`: Khóa ngoại đến bảng `users` (id).
    - `address_id`: Khóa ngoại đến bảng `addresses` (id).

- `fulfilled_order_id`: Khóa ngoại đến bảng `fulfilled_orders` (id), có thể là null.
- `created_at`: Thời điểm tạo, kiểu timestamp, không được để trống.
- `updated_at`: Thời điểm cập nhật, kiểu timestamp, không được để trống.
- `deleted_at`: Thời điểm xóa, kiểu timestamp, có thể là null.

Bảng `order_items`:

- Chức năng: Lưu trữ thông tin về các mục hàng trong đơn hàng.
- Các cột:
  - `id`: Khóa chính, số nguyên không dấu, tự động tăng.
  - `product_id`: Khóa ngoại đến bảng `products` (id).
  - `quantity`: Số nguyên không dấu, không được để trống.
  - `price`: Số thực không dấu, không được để trống.
  - `order_id`: Khóa ngoại đến bảng `orders` (id).

Bảng `payment_methods`:

- Chức năng: Lưu trữ thông tin về các phương thức thanh toán.
- Các cột:
  - `id`: Khóa chính, số nguyên không dấu, tự động tăng.
  - `name`: `varchar(255)`, không được để trống.
  - `code`: `varchar(255)`, không được để trống.
  - `enable`: Số nguyên nhỏ (1 hoặc 0), không được để trống.

Bảng `products`:

- Chức năng: Lưu trữ thông tin về các sản phẩm.
- Các cột:
  - `id`: Khóa chính, số nguyên không dấu, tự động tăng.
  - `name`: `varchar(255)`, không được để trống.
  - `description`: Kiểu văn bản, có thể là null.
  - `price`: Số thực không dấu, không được để trống.

- `sku`: `varchar(255)`, không được để trống.
- `availability`: Số nguyên nhỏ (1 hoặc 0), không được để trống.
- `quantity`: Số nguyên, có thể là null.
- `discount_price`: Số thực không dấu, có thể là null.
- `slug`: `varchar(255)`, không được để trống.
- `created_at`: Thời điểm tạo, kiểu timestamp, không được để trống.
- `updated_at`: Thời điểm cập nhật, kiểu timestamp, không được để trống.
- `deleted_at`: Thời điểm xóa, kiểu timestamp, có thể là null.

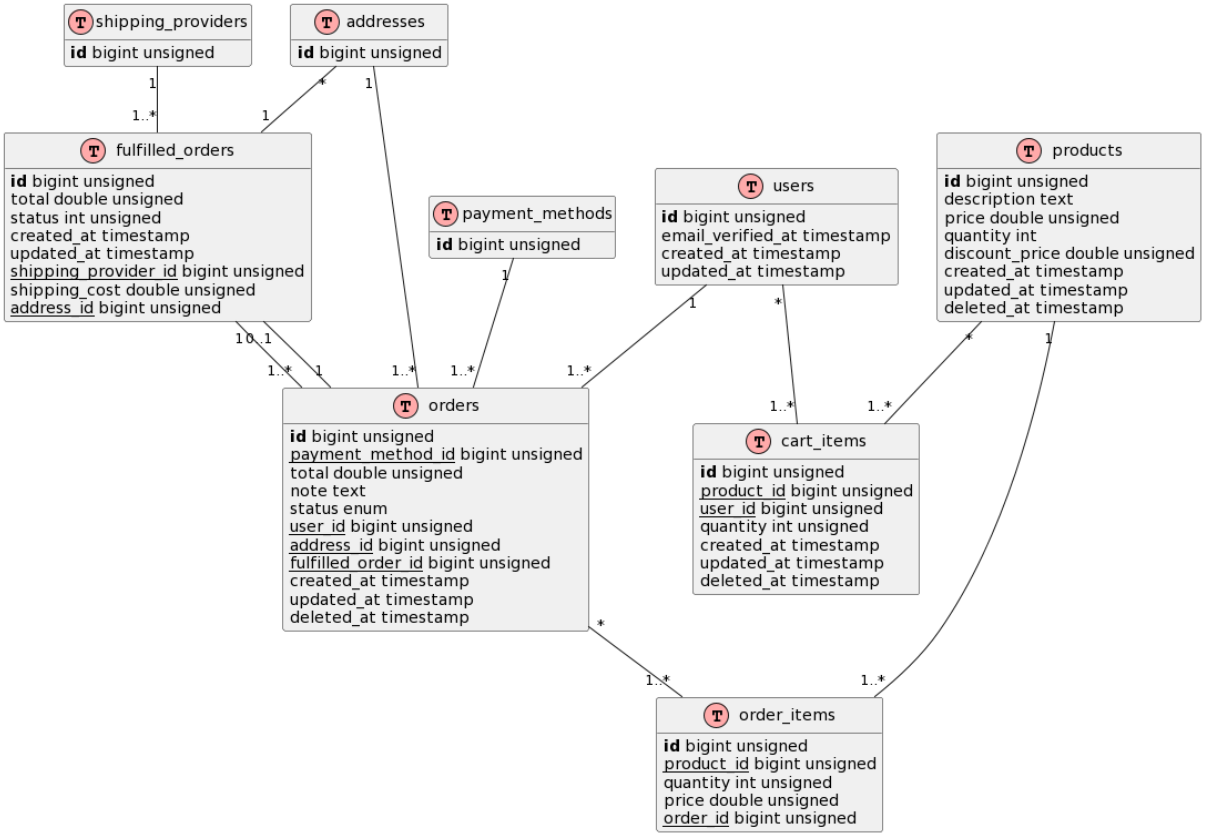
Bảng `shipping_providers`:

- Chức năng: Lưu trữ thông tin về nhà cung cấp vận chuyển.
- Các cột:
  - `id`: Khóa chính, số nguyên không dấu, tự động tăng.
  - `name`: `varchar(255)`, không được để trống.
  - `code`: `varchar(255)`, không được để trống.
  - `enable`: Số nguyên nhỏ (1 hoặc 0), không được để trống.

Bảng `users`:

- Chức năng: Lưu trữ thông tin về người dùng.
- Các cột:
  - `id`: Khóa chính, số nguyên không dấu, tự động tăng.
  - `name`: `varchar(255)`, không được để trống.
  - `email`: `varchar(255)`, không được để trống, duy nhất.
  - `email_verified_at`: Thời điểm xác minh email, kiểu timestamp, có thể là null.
  - `password`: `varchar(255)`, không được để trống.
  - `remember_token`: `varchar(100)`, có thể là null.
  - `created_at`: Thời điểm tạo, kiểu timestamp, không được để trống.
  - `updated_at`: Thời điểm cập nhật, kiểu timestamp, không được để trống.

Những bảng có cột `deleted_at` cho thấy có sử dụng cơ chế "soft delete" (soft delete là phương pháp đánh dấu thời gian thực hiện xóa bản ghi ở cột `deleted_at` nhưng không xóa bản ghi trong cơ sở dữ liệu, website sẽ dựa vào đó để truy vấn và chỉ trả về những bản ghi chưa có giá trị `deleted_at`).



Hình 19: Biểu đồ quan hệ mô tả cấu trúc các bảng và quan hệ giữa chúng trong cơ sở dữ liệu của website TMĐT

3.3. Xây dựng giao diện và trải nghiệm người dùng

3.3.1. Cài đặt và sử dụng thư viện Tailwind

Tác giả sử dụng hệ thống thiết kế cơ bản của Tailwind làm hệ thống thiết kế chính cho dự án. Tailwind là một thư viện design component phổ biến trong cộng đồng do đó có đa dạng thiết kế và ý tưởng được hỗ trợ từ cộng đồng. Kèm theo đó là sử dụng công cụ Vite để biên dịch và đóng gói code [4].

Để cài đặt Tailwind cho website ta thực hiện các bước sau:

Cài đặt thư viện Tailwind bằng câu lệnh:

```
npm install -D tailwindcss
```

Khởi tạo file cấu hình Tailwind bằng câu lệnh:

```
npx tailwindcss init
```

Sau khi chạy câu lệnh hệ thống sẽ tạo ra file `tailwind.config.js` với nội dung bên dưới, trong file này ta có thể cài đặt thêm các plugin, theme và khai báo vào thuộc tính tương ứng, ngoài ra ta còn có thể cấu hình cho Tailwind tự động đọc và hot-reload những file được khai báo trong thuộc tính `content`:

```
export default {
    content: [],
    corePlugins: {},
    plugins: [],
};
```

Sau khi tác giả cài đặt các thư viện cần thiết và khai báo đường dẫn những file giao diện web thì file có nội dung như sau:

```
import forms from '@tailwindcss/forms';

/** @type {import('tailwindcss').Config} */
export default {
    content: [
        './vendor/laravel/framework/src/Illuminate/Pagination/resources/views/
*.blade.php',
        './storage/framework/views/*.php',
        './resources/views/**/*.blade.php',
    ],
    corePlugins: {
        aspectRatio: false,
    },
    plugins: [forms, require('@tailwindcss/typography'), require('@tailwindcss/aspect-
ratio'), require('@tailwindcss/forms')],
};
```

Tiếp đến ta tạo file `app.css` với nội dung như bên dưới, file CSS này import các thư viện CSS của Tailwind:

```
@tailwind base;
@tailwind components;
@tailwind utilities;
```

Trong file `vite.config.js` ta cấu hình Tailwind cho Laravel như sau:

```
import { defineConfig } from 'vite';
import laravel from 'laravel-vite-plugin';
export default defineConfig({
    plugins: [
        laravel({
            input: ['path/to/app.css'], // replace your app.css path
            refresh: true,
        }),
    ],
});
```

Sau khi đã cài đặt và cấu hình, ta chạy câu lệnh bên dưới để Tailwind tiến hành đọc và biên dịch những file được khai báo:

```
npm run dev
```

Cuối cùng mở Visual Studio Code và truy cập mã nguồn giao diện của website để lập trình. Ví dụ một đoạn mã giao diện sử dụng các class của thư viện Tailwind trong Laravel Blade:

```
<div {{ $attributes->merge(['class' => 'rounded-md bg-green-50 dark:bg-green-900 p-4']) }}
>
    <div class="flex">
        <div class="flex-shrink-0">
            <!-- Heroicon name: solid/check-circle -->
            <svg class="h-5 w-5 text-green-400 dark:text-green-200" xmlns="http://www.w3.
org/2000/svg" viewBox="0 0 20 20" fill="currentColor" aria-hidden="true">
                <path fill-rule="evenodd" d="M10 18a8 8 0 100-16 8 8 0 000
16zm3.707-9.293a1 1 0 00-1.414-1.414L9 10.586 7.707 9.293a1 1 0 00-1.414 1.414l2 2a1 1 0
001.414 0l4-4z" clip-rule="evenodd" />
            </svg>
        </div>
        <div class="ml-3">
            <p class="text-sm font-medium text-green-800 dark:text-green-200">
                {{ $slot }}
            </p>
        </div>
    </div>
</div>
```

Kết quả hiển thị:

Hình 20: Giao diện thông báo thành công của website

3.3.2. Xây dựng thành phần giao diện website

Thành phần giao diện của website được xây dựng trong thư mục `views` của website bao gồm các thư mục chính bên trong sau:

- `components`: Chứa các thành phần giao diện có thể tái sử dụng nhiều lần trong website như giao diện thông báo, bảng, ô nhập, menu,…
- `layouts`: Chứa các khung giao diện có thể tái sử dụng nhiều lần ở các trang trong website. Ví dụ như khung giao diện quản trị nội dung và khung giao diện website dành cho khách hàng.
- `livewire`: Chứa những thành phần giao diện website có thể tương tác trong thời gian thực với server như trang quản lý giỏ hàng, nút thêm sản phẩm vào giỏ hàng.
- `admin`: Chứa các giao diện của các trang quản trị website.

Việc phân tách các giao diện có chức năng tương tự nhau vào các thư mục cụ thể giúp cho việc quản lý mã nguồn giao diện dễ dàng hơn và tối ưu hiệu suất khi lập trình.

3.3.3. Trải nghiệm người dùng

Tác giả đã dành nhiều thời gian tham gia trải nghiệm các sản phẩm tương tự khác, trong số đó có nhiều sản phẩm phổ biến để đánh giá ưu nhược điểm của trải nghiệm người dùng từ đó cải thiện trải nghiệm cho sản phẩm này.

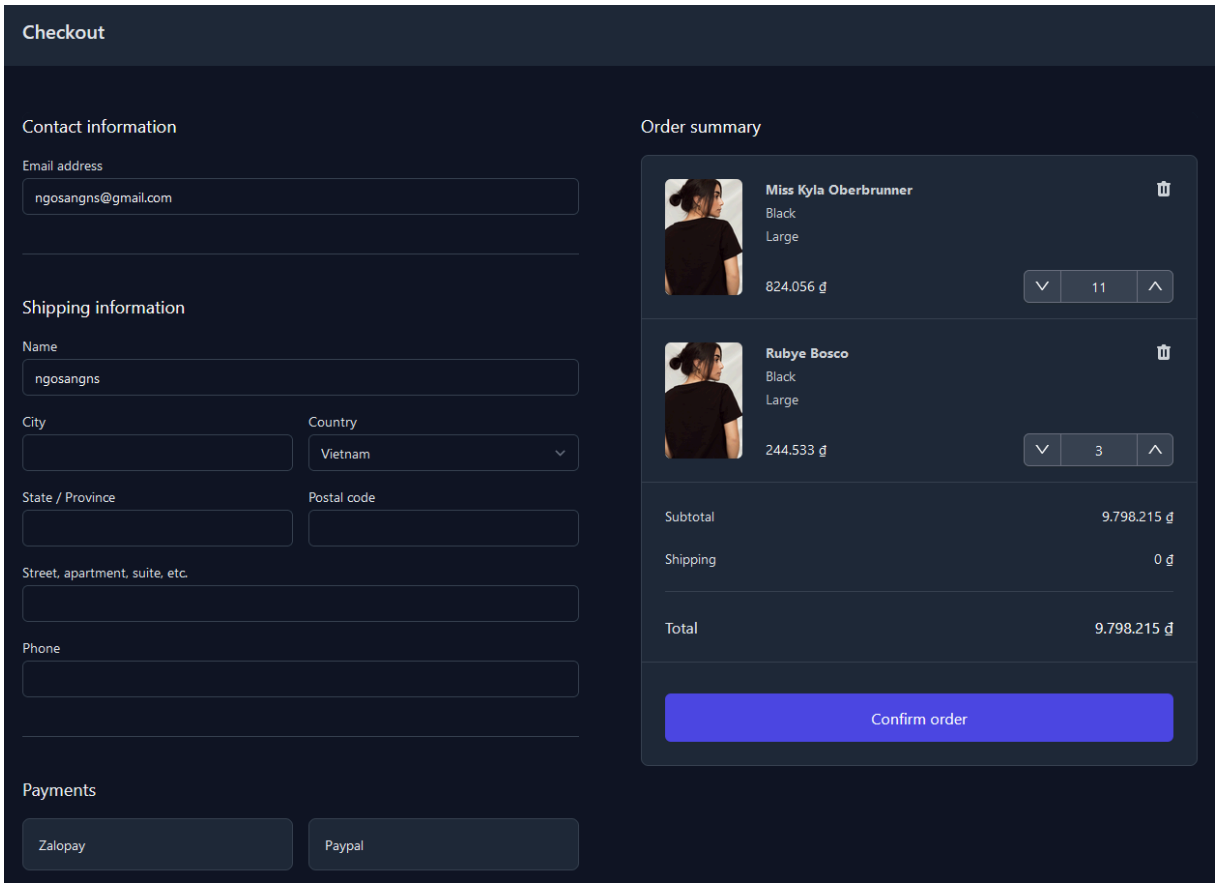
Trong quá trình xây dựng giao diện và trải nghiệm người dùng cho website TMĐT, tác giả đã đặt trọng điểm vào việc mang lại trải nghiệm tuyệt vời và thuận tiện cho người dùng. Đây là một phần quan trọng để thu hút và giữ chân khách hàng. Đảm bảo người dùng dễ dàng tìm thấy và khám phá các sản phẩm một cách

nhanh chóng. Giao diện được thiết kế đơn giản, với bố cục rõ ràng và các thành phần giao diện được sắp xếp hợp lý, tạo ra một trải nghiệm trực quan và dễ sử dụng.

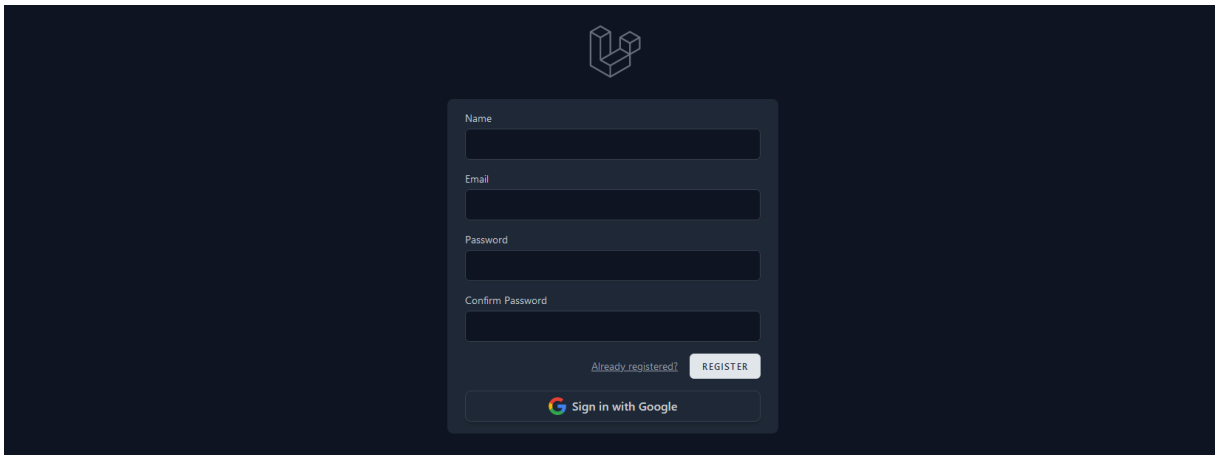
Việc đăng ký tài khoản cũng được đơn giản hóa để người dùng có thể nhanh chóng trở thành thành viên của trang web. Website đã tích hợp các tùy chọn đăng ký thông qua email và mật khẩu, cùng với việc đăng ký bằng tài khoản Google, giúp người dùng tiết kiệm thời gian hơn.

Tính năng giỏ hàng được thiết kế để giúp người dùng quản lý và kiểm soát quá trình mua hàng một cách thuận tiện. Người dùng có thể dễ dàng thêm và xóa sản phẩm trong giỏ hàng, cập nhật số lượng sản phẩm và xem tổng giá trị đơn hàng. Điều này giúp người dùng có trải nghiệm mua sắm trực tuyến mượt mà và tiết kiệm thời gian.

Website cũng đã tích hợp nhiều phương thức thanh toán đa dạng để đáp ứng nhu cầu và sự thuận tiện của người dùng. Các tùy chọn thanh toán như Zalopay, Paypal và nhiều hơn nữa được tích hợp vào website, cho phép người dùng lựa chọn phương thức phù hợp và hoàn tất quá trình thanh toán một cách dễ dàng.

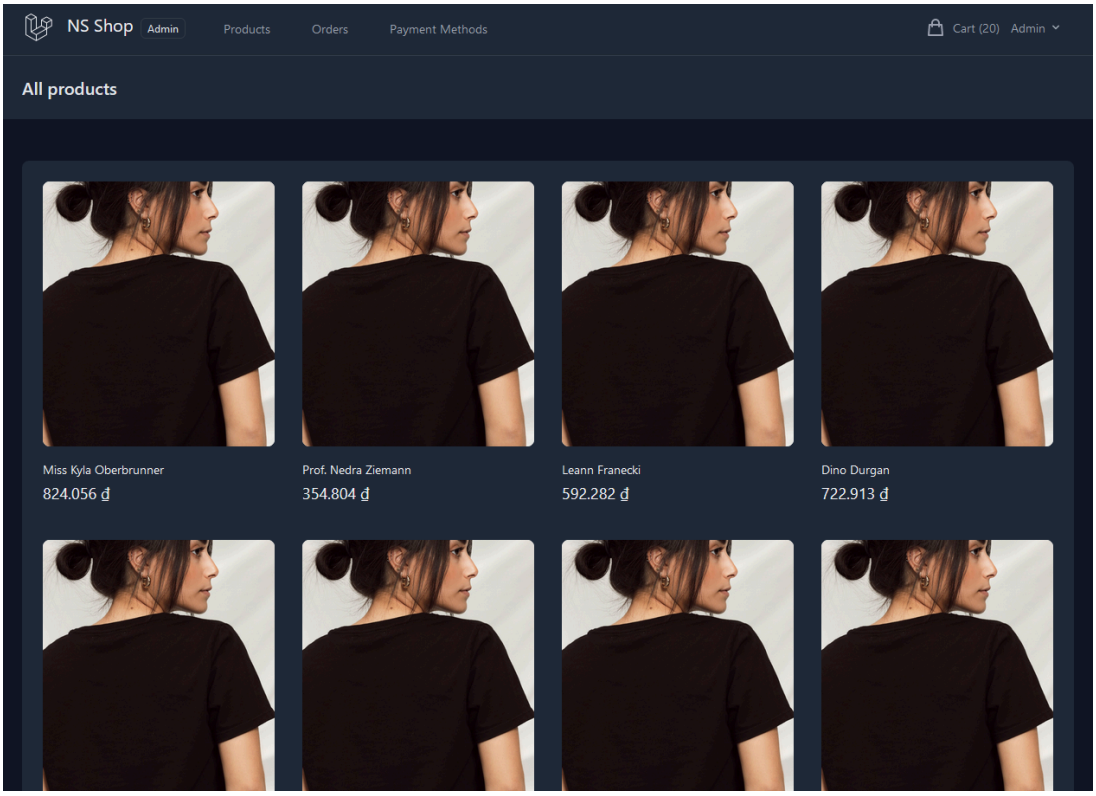


Hình 21: Giao diện trang xem giỏ hàng của website

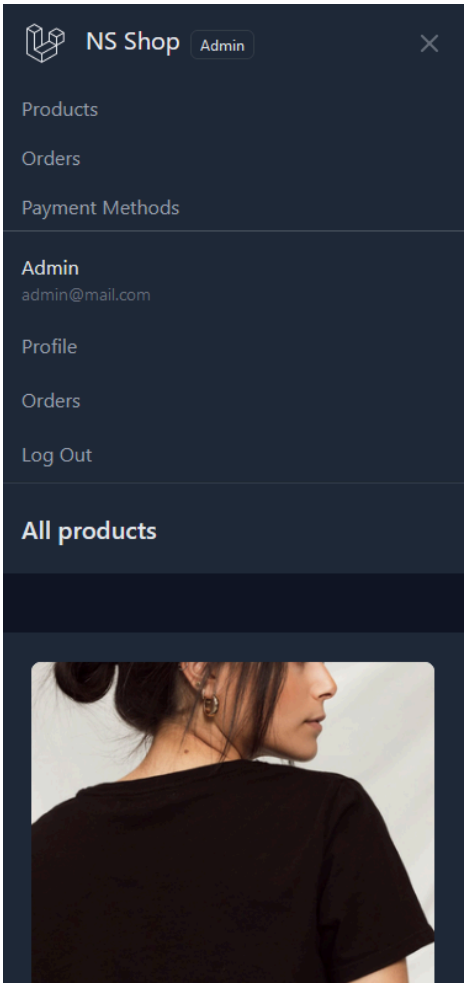


Hình 22: Giao diện trang đăng ký của website

Website cũng đã được tối ưu hóa cho trải nghiệm di động, giúp người dùng truy cập và duyệt sản phẩm trên các thiết bị di động một cách thuận tiện. Thiết kế đáp ứng (RWD) đã được áp dụng để đảm bảo giao diện hiển thị tốt trên các kích thước màn hình khác nhau, từ điện thoại thông minh đến máy tính bảng và máy tính để bàn.



Hình 23: Giao diện website khi responsive ở màn hình máy tính (1280px)



Hình 24: Giao diện website khi responsive ở màn hình Iphone 11 Pro (375px)

3.4. Lập trình các chức năng và tính năng

3.4.1. Tính năng đăng nhập và đăng ký với email/password

Mặc định website sẽ sử dụng email/password để tiến hành đăng nhập, website sẽ lưu thông tin tên, email của người dùng. Ngoài ra còn hỗ trợ tính năng "Remember me" giúp người dùng dễ dàng quản lý phiên đăng nhập cho mình. [5]

Code xử lý tính năng đăng nhập với email/password:

```
if (! Auth::attempt($this->only('email', 'password'), $this->boolean('remember'))) {
    RateLimiter::hit($this->throttleKey());

    throw ValidationException::withMessages([
        'email' => trans('auth.failed'),
```

```
    ]);
}
```

Code xử lý tính năng đăng ký với email/password:

```
public function store(Request $request): RedirectResponse
{
    $request->validate([
        'name' => ['required', 'string', 'max:255'],
        'email' => ['required', 'string', 'email', 'max:255', 'unique:' . User::class],
        'password' => ['required', 'confirmed', 'min:8', Rules\Password::defaults()],
    ]);
    $user = User::create([
        'name' => $request->name,
        'email' => $request->email,
        'password' => Hash::make($request->password),
    ]);
    event(new Registered($user));
    Auth::login($user);
    return redirect(RouteServiceProvider::HOME);
}
```

3.4.2. Tính năng đăng nhập và đăng ký với tài khoản Google

Tính năng đăng nhập và đăng ký với tài khoản Google sử dụng chuẩn OAuth 2.0 để triển khai trong website. Các bước triển khai được trình bày bên dưới. [6]

Tạo Google OAuth Credentials:

- Vào trang https://console.developers.google.com và tạo project mới.
- Bật tính năng Google+ API bằng cách vào mục "Library" và tìm "Google+ API". Chọn và kích hoạt.
- Vào mục "Credentials" và chọn "Create Credentials". Chọn "OAuth client ID" ở menu xổ xuống.
- Cấu hình "name" và "authorized domains" của mục OAuth consent screen.
- Chọn "Web application".
- Thêm authorized redirect URIs. Ví dụ ở đây là `http://localhost/auth/google/callback` cho môi trường phát triển.
- Chọn "Create" để tạo OAuth client. Lưu giá trị mục "Client ID" và "Client Secret" vừa tạo.

Cấu hình Laravel:

Mở file `.env` và thêm Google OAuth client credentials:

```
GOOGLE_CLIENT_ID=your-client-id
GOOGLE_CLIENT_SECRET=your-client-secret
GOOGLE_REDIRECT_URI=http://localhost/auth/google/callback
```

Tạo routes cho việc xác thực Google ở file `routes/web.php`:

```
Route::get('/auth/google', [LoginController::class, 'redirectToGoogle']);
Route::get('/auth/google/callback', [LoginController::class, 'handleGoogleCallback']);
```

Tạo mới controller tên `LoginController` sử dụng câu lệnh sau:

```
php artisan make:controller Auth/LoginController
```

Mở file `LoginController.php` và triển khai hàm `redirectToGoogle` có chức năng chuyển hướng người dùng đến trang xác thực của Google và hàm `handleGoogleCallback` có chức năng nhận dữ liệu đã xác thực phía Google gửi về:

```
public function redirectToGoogle()
{
    $params = [
        'client_id' => config('app.google_client_id'),
        'redirect_uri' => config('app.google_redirect_uri'),
        'response_type' => 'code',
        'scope' => 'openid email profile',
        'state' => csrf_token(),
    ];
    $url = 'https://accounts.google.com/o/oauth2/auth?' . http_build_query($params);
    return redirect($url);
}

public function handleGoogleCallback(Request $request)
{
    $state = $request->query('state');
    $code = $request->query('code');

    if ($state !== csrf_token())
        return redirect('/login')->withErrors('Invalid state parameter');

    $response = Http::asForm()->post('https://oauth2.googleapis.com/token', [
```

```
        'code' => $code,
        'client_id' => config('app.google_client_id'),
        'client_secret' => config('app.google_client_secret'),
        'redirect_uri' => config('app.google_redirect_uri'),
        'grant_type' => 'authorization_code',
    ]);

    if ($response->failed())
        return redirect('/login')->withErrors('Failed to retrieve access token');

    $access_token = $response->json('access_token');

    $response = Http::withHeaders([
        'Authorization' => 'Bearer ' . $access_token,
    ])->get('https://www.googleapis.com/oauth2/v3/userinfo');

    if ($response->failed())
        return redirect('/login')->withErrors('Failed to retrieve user information');

    $user = $response->json();

    Auth::loginUsingId($userId);

    return redirect('/home');
}
```

Đoạn code bên trên cũng là code triển khai OAuth 2.0 của Google vào website Laravel.

Cuối cùng thêm đoạn code hiển thị nút đăng nhập với Google vào trang đăng nhập của website:

```
<div {{ $attributes }}>
    <a href="{{ route('auth.google.redirect') }}"
        class="flex items-center justify-center bg-white dark:bg-gray-800 text-gray-700
dark:text-gray-300 font-semibold py-2 px-4 border border-gray-300 dark:border-gray-700
rounded-lg shadow-md transition duration-300 ease-in-out hover:bg-gray-100 dark:hover:bg-
gray-700 hover:border-gray-400 dark:hover:border-gray-600 focus:outline-none focus:ring-2
focus:ring-offset-2 focus:ring-blue-500 dark:focus:ring-offset-gray-800">
        <img class="w-6 h-6 mr-2" src="{{ asset('assets/images/google.svg') }}" alt="Login
with Google" />
        Sign in with Google
    </a>
</div>
```

Sử dụng tính năng:

- Khởi động Docker, sau đó khởi Laravel bằng câu lệnh:

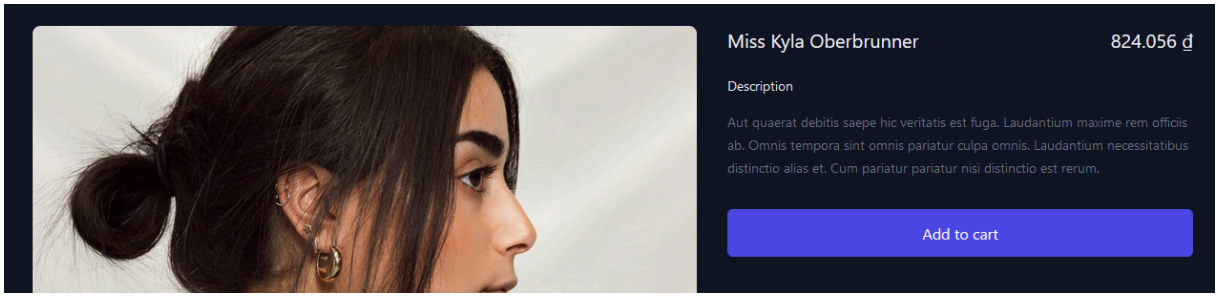
```
sail up -d
```

- Vào link trang đăng nhập của website sau đó nhấn chọn "Login with Google".
- Ta sẽ được chuyển hướng đến trang đăng nhập của Google. Tiến hành đăng nhập và xác thực tài khoản.
- Sau khi xác thực thành công ta sẽ được chuyển hướng về trang `/home`.

3.4.3. Tính năng giỏ hàng

Tính năng giỏ hàng của website yêu cầu khách hàng phải đăng nhập để sử dụng. Dựa vào mô hình dữ liệu được thiết kế ở mục 2.2.2 thì mỗi khách hàng sẽ có một giỏ hàng duy nhất.

Sau khi đăng nhập, khi khách hàng nhấn nút "Add to cart" thì sản phẩm sẽ được thêm vào giỏ hàng của khách hàng và được cập nhật ngay lập tức mà không cần load lại trang. Việc không load lại trang giúp cho trải nghiệm mua sắm của khách hàng được mượt mà trơn tru hơn.



Hình 25: Giao diện của nút thêm sản phẩm vào giỏ hàng

Code giao diện của tính năng thêm sản phẩm vào giỏ hàng:

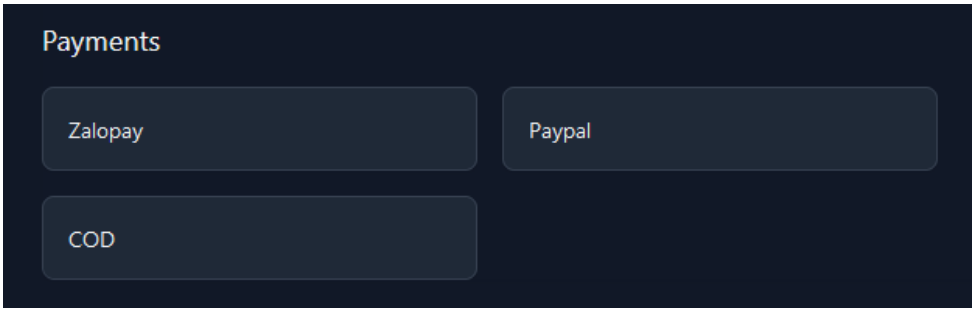
```
<button wire:loading.attr="disabled" wire:target="addToCart" wire:click="addToCart"
    class="mt-8 w-full bg-indigo-600 border border-transparent rounded-md py-3 px-8 flex
items-center justify-center text-base font-medium text-white hover:bg-indigo-700
focus:outline-none focus:ring-2 focus:ring-offset-2 focus:ring-indigo-500">
    <x-loading-spin class="mt-2" wire:loading wire:target="addToCart" />
    Add to cart
</button>
```

Code xử lý của tính năng thêm sản phẩm vào giỏ hàng:

```
public function addToCart($quantity = 1) {
    try {
        $user = Auth::user();

        if (!$user->id)
            throw new Exception('You must be logged in to add to cart.', 1);

        // check quantity
        if ($quantity < 1)
            throw new Exception('Quantity must be greater than 0.', 1);

        // check product exists
        $product = Product::find($this->productId);
        if (is_null($product))
            throw new Exception('Product not found.', 1);

        // add to cart
        $cartItem = CartItem::where('user_id', $user->id)
            ->where('product_id', $product->id)->first();
        if (is_null($cartItem)) {
            CartItem::create([
                'product_id' => $product->id,
                'quantity' => $quantity,
                'user_id' => $user->id,
            ]);
        } else {
            $cartItem->quantity = $cartItem->quantity + $quantity;
            $cartItem->save();
        }

        // response
        $this->alert('success', 'Product added successfully');
        $this->emit("cart-update");
    } catch (Exception $e) {
        $this->alert(
            'error',
            $e->getCode() ? $e->getMessage() : "Error updating quantity."
        );
    }
}
```

3.4.4. Tính năng thanh toán

Website có 3 phương thức thanh toán chính: Thanh toán qua Zalopay, Paypal và thanh toán Cash On Delivery (COD).



Hình 26: Giao diện chọn phương thức thanh toán trong website

Thông tin người dùng cần nhập khi thanh toán bao gồm: email người nhận, tên người nhận, số điện thoại người nhận và địa chỉ.

Code kiểm tra thông tin người dùng nhập vào đơn hàng, nếu thông tin đúng format và chính xác thì sẽ tạo đơn hàng:

```
protected $rules = [
    'name' => 'required',
    'email' => 'required|email',
    'country' => 'required',
    'city' => 'required',
    'state' => 'required',
    'street' => 'required',
    'postalCode' => 'required|postal_code',
    'phone' => 'required|vn_phone_number',
    'paymentMethodId' => 'required',
    'cart' => 'required|array|min:1',
];

// ...

public function createOrder() {
    $this->validate();

    try {
        $user = $this->getUser();

        // custom validate
        if (is_null(PaymentMethod::where('id', $this->paymentMethodId)
            ->where('enable', true)->first()))
            throw new Exception('Payment method does not exist.', 1);

        $count = $user->cartItems()
            ->whereIn('id', array_map(fn ($item) => $item['id'], $this->cart))->count();
        if ($count != count($this->cart))
            throw new Exception('Cart items are wrong.', 1);
```

```
        $productIds = array_map(fn ($item) => $item['product_id'], $this->cart);
        $count = Product::whereIn('id', $productIds)->count();
        if ($count != count($productIds))
            throw new Exception('Cart items are wrong.', 1);

        foreach ($this->cart as &$item)
            if ($item['user_id'] != $user->id || $item['quantity'] < 1)
                throw new Exception('Cart items are wrong.', 1);

        DB::beginTransaction();
        $addressId = Address::insertGetId([
            'country' => $this->country,
            'city' => $this->city,
            'state' => $this->state,
            'street' => $this->street,
            'zip_code' => $this->postalCode,
        ]);
        $orderId = Order::insertGetId([
            'user_id' => $user->id,
            'address_id' => $addressId,
            'name' => $this->name,
            'email' => $this->email,
            'phone' => $this->phone,
            'total' => $this->total(),
            'payment_method_id' => $this->paymentMethodId,
            'status' => OrderStatus::UNPAID,
        ]);
        OrderItem::insert(
            array_map(function ($v) use ($orderId) {
                return [
                    'order_id' => $orderId,
                    'product_id' => $v['product']['id'],
                    'quantity' => $v['quantity'],
                    'price' => $v['product']['price'],
                ];
            }, $this->cart)
        );
        $user->cartItems()->delete();
        DB::commit();
        return redirect()->route('order.pay', ['order' => $orderId]);
    } catch (Exception $e) {
        DB::rollBack();
        return $this->handleException($e);
    }
}
```

Code xử lý thanh toán:

```
```

```
public function pay(Request $request, Order $order)
{
    $order->with('paymentMethod');

    // cod
    if ($order->paymentMethod->code === 'cod')
        return redirect()->route('order.show', $order->id);

    // process payment
    $paymentGateway = null;
    switch ($order->paymentMethod->code) {
        case 'zalopay':
            $paymentGateway = new Zalopay();
            break;
        case 'paypal':
            $paymentGateway = new Paypal();
            break;
    }
    return $paymentGateway->pay($order);
}
```

3.4.5. Tính năng quản lý đơn hàng

Mỗi người dùng chỉ quản lý đơn hàng của mình.

Quản trị viên có thể xem tất cả đơn hàng.

Code xử lý lấy dữ liệu đơn hàng của người dùng:

```
public function show(Order $order)
{
    $order->with('paymentMethod', 'address', 'orderItems.product');
    return view('orders.show', ['order' => $order]);
}
```

3.5. Kết chương

Qua chương 3 tác giả đã trình bày các bước triển khai xây dựng các tính năng chính trong một website TMĐT dựa vào các khảo sát phân tích và những thiết kế mô hình được thực hiện ở Chương 1 và Chương 2.

KẾT LUẬN CHUNG

Thương mại điện tử mở ra cơ hội kinh doanh trực tuyến trên toàn cầu. Khách hàng có thể truy cập và mua hàng từ bất kỳ đâu và bất kỳ khi nào, giúp mở rộng phạm vi tiếp cận khách hàng và tăng doanh thu. Doanh nghiệp ngày nay cần cạnh tranh trên không gian trực tuyến để tồn tại và phát triển. Website thương mại điện tử cho phép doanh nghiệp tiếp cận được khách hàng tiềm năng và cung cấp thông tin sản phẩm, dịch vụ, đánh giá, và khuyến mãi để thu hút và giữ chân khách hàng cũng như mở rộng khả năng tiếp cận thị trường, giúp doanh nghiệp tăng doanh số bán hàng và khách hàng. Do đó việc xây dựng website thương mại điện tử là một công việc có tính cấp thiết cao trong thời đại số hóa ngày nay. Đề tài này đã giúp tác giả nhận thức rõ hơn về tính cấp thiết đó và tạo động lực để tiếp tục nghiên cứu và phát triển trong lĩnh vực này.

Qua quá trình thực hiện đề tài nghiên cứu này, đồ án đã đạt được một số mục tiêu mong muốn. Tác giả đã tiến hành phân tích, khảo sát và xây dựng và hoàn thiện phần lớn đề tài, bao gồm xác định các yêu cầu chức năng và yêu cầu an toàn, thực hiện việc phân tích thiết kế các yêu cầu bằng các biểu đồ phân rã chức năng, biểu đồ use-case, mô hình quan hệ và lưu trữ dữ liệu,… cho đến việc triển khai các chức năng quan trọng của website TMĐT như tính năng đăng nhập/đăng ký, giỏ hàng và thanh toán. Tạo nền tảng cho việc phát triển website TMĐT trong thực tế.

Tuy nhiên, do thời gian nghiên cứu, kiến thức và tìm hiểu còn hạn chế do đó đồ án còn một số thiếu sót. Tác giả nhận thức được rằng đề tài này đòi hỏi sự hiểu biết và kỹ năng rộng hơn để đạt được một hệ thống hoàn thiện.

Trong tương lai, tác giả sẽ tiếp tục nghiên cứu, tìm hiểu và nâng cấp hệ thống để hoàn thiện và đáp ứng được yêu cầu sử dụng trong thực tế. Tác giả rất mong nhận được ý kiến đóng góp quý báu từ thầy cô để đồ án có thể phát triển và hoàn thiện hơn nữa.

Để phát triển và nâng cao chất lượng hệ thống, tác giả sẽ tiếp tục nghiên cứu, tìm hiểu và áp dụng các công nghệ mới, cũng như thực hiện các bước nâng cấp và tối ưu hóa hệ thống hiện tại.

TÀI LIỆU THAM KHẢO

[1] “Những điểm nổi bật của thương mại điện tử Việt Nam trong giai đoạn tới.” [Online]. Available: https://tapchicongthuong.vn/bai-viet/nhung-diem-noi-bat-cua-thuong-mai-dien-tu-viet-nam-trong-giai-doan-toi-104548.htm

[2] “OWASP Top 10 Vulnerabilities 2021.” [Online]. Available: https://www.edudwar.com/owasp-top-10-vulnerabilities

[3] TS. Nguyễn Tuấn Anh and KS. Hoàng Thanh Nam, *Giáo trình xây dựng web an toàn*. 2013.

[4] “Vite: Build tool front-end 'thế hệ mới'.” [Online]. Available: https://atekco.io/1663226080655-vite-build-tool-front-end-the-he-moi

[5] “Laravel Breeze.” [Online]. Available: https://github.com/laravel/breeze

[6] “Using OAuth 2.0 for Web Server Applications.” [Online]. Available: https://developers.google.com/identity/protocols/oauth2/web-server?hl=en