Bộ Giáo Dục Và Đào Tạo

Trường Đại Học Ngoại Ngữ - Tin Học Thành Phố Hồ Chí Minh

Khoa Công Nghệ Thông Tin



MÔN HỌC : THƯƠNG MẠI ĐIỆN TỬ

ĐỀ TÀI : Web Bán Tiền

Giáo Viên Hướng Dẫn : Tưởng Thị Xuân Thu

Thành Viên :

|  |  |
| --- | --- |
| 1. Nguyễn Thị Thanh Ngân | MSSV: 20DH111120 |
| 1. Mai Thị Quỳnh Như | MSSV: 20DH110371 |

*Tp. Hồ Chí Minh, Ngày tháng 12 năm 2022*

PHIẾU CHẤM ĐIỂM MÔN THI VẤN ĐÁP

Điểm phần trình bày – Điểm hệ 10 – Tỷ lệ điểm chiếm 30%



|  |  |  |
| --- | --- | --- |
|  | CBCT1 | CBCT2 |
| Họ tên  CBCT | …………………………………………..  Chữ ký: ……………………………....... | …………………………………………..  Chữ ký: ……………………………....... |
| Điểm | …………………………………………..  Bằng chữ: ……………………………... | …………………………………………..  Bằng chữ: ……………………………... |
| Nhận xét |  |  |

Điểm quá trình – Điểm hệ 10 – Tỷ lệ điểm chiếm 70%



Họ tên CBCT: …………………………………………………………………………………

Điểm tổng kết: …………………(Bằng chữ:…………………………………………)



Nhận xét của giảng viên

http://vuatien-bantien.somee.com

Lời cảm ơn

Đầu tiên nhóm em xin gửi lời cảm ơn đến cô Tưởng Xuân Thu-Người đã dạy chúng em môn Thương Mại Điện Tử trong kì vừa rồi. Cùng với sự hướng đã của cô đã giúp chúng em hoàng thành và mang lại cho bản thân rất nhiều kiến thức về môn học mới này.

Bên cạnh đó nhóm em cũng không quên gửi lời cảm ơn đến những thầy cô trong các bộ môn khác và các thầy cô trong trường đã nhiệt tình giảng dạy cho chúng em trong thời gian qua. Những kiến thức đó đã giúp chúng em hoàn thành tốt đồ án này.

Bài thiết kế web bán Tiền là kết quả của quá trình cố gắng không ngừng của nhóm chúng em trong 2 tháng qua. Tuy nhiên vì còn vài hạn chế và những kiến thức mới chúng em không thể tránh khỏi những thiếu sót.

Chúng em rất mong nhận được lời đóng góp từ quý thầy cô.

Phụ Lục

[I. Giới thiệu chung 9](#_Toc121805698)

[1. Giới thiệu nguyên nhân có hệ thống 9](#_Toc121805699)

[2. Các khó khăn khi không có hệ thông 9](#_Toc121805700)

[3. Các bất cập của hệ thống 9](#_Toc121805701)

[4. Lợi ích của hệ thống mang lại 9](#_Toc121805702)

[5. Yêu cầu chức năng 10](#_Toc121805703)

[6. Yêu cầu phi chức năng 10](#_Toc121805704)

[II. Phân tích thiết kế hệ thống 10](#_Toc121805705)

[1. Đặc tả hệ thống (System requirements) 10](#_Toc121805706)

[2. Giả định (Assumption) cho hệ thống 11](#_Toc121805707)

[3. Phạm vi của hệ thống (Scope) 12](#_Toc121805708)

[4. Yêu cầu của người sử dụng hệ thống (Users requirements) 13](#_Toc121805709)

[4.1.Yêu cầu chức năng 13](#_Toc121805710)

[4.2.Yêu cầu phi chức năng 13](#_Toc121805711)

[5. Ma trận actor và use case 13](#_Toc121805712)

[5.1. Ma trận Actor 13](#_Toc121805713)

[5.2. Use case 14](#_Toc121805714)

[6. Actor và use case 15](#_Toc121805719)

[6.1. Đặc tả use case 15](#_Toc121805720)

[6.2. Activity diagram 33](#_Toc121805736)

[6. Sơ đồ ERD 42](#_Toc121805763)

[7. Giao diện 42](#_Toc121805765)

[7.1. Giao diện admin 42](#_Toc121805766)

[7.2. Giao diện user 44](#_Toc121805771)

[8. Code 48](#_Toc121805781)

[8.1. Code admin 48](#_Toc121805782)

[8.2. Code user 61](#_Toc121805786)

[9. Phân tích CSDL 84](#_Toc121805792)

[9.1.Query 84](#_Toc121805793)

[III. Kết luận 93](#_Toc121805794)

[IV. Tài liệu tham khảo 93](#_Toc121805805)

Danh mục hình ảnh

[Hình 1: Usecase khách hàng 19](#_Toc121785697)

[Hình 2: Usecase quản lí 20](#_Toc121785699)

[Hình 3: Activity Diagram tìm kiếm 38](#_Toc121785719)

[Hình 4: Activity Diagram Thêm giỏ hàng 38](#_Toc121785721)

[Hình 5: Activity Diagram xem giỏ hàng 39](#_Toc121785723)

[Hình 6: Activity Diagram đặt hàng 39](#_Toc121785725)

[Hình 7: Activity Diagram thanh toán 40](#_Toc121785727)

[Hình 8: Activity Diagram bình luận sản phẩm 41](#_Toc121785729)

[Hình 9: Activity Diagram duyệt đơn hàng 41](#_Toc121785731)

[Hình 10: Activity Diagram xóa sản phẩm 42](#_Toc121785733)

[Hình 11: Activity Diagram lọc sản phẩm 42](#_Toc121785735)

[Hình 12: xem chi tiết sản phẩm 43](#_Toc121785737)

[Hình 13: Activity Diagram Thêm sảm phẩm 43](#_Toc121785739)

[Hình 14: Activity Diagram sửa sản phẩm 44](#_Toc121785741)

[Hình 15: Activity Diagram thêm nhân viên 45](#_Toc121785743)

[Hình 16: ERD 46](#_Toc121785745)

[Hình 18: Giao diện quản lý sản phẩm 46](#_Toc121785748)

[Hình 18: Giao diện quản lí khách hàng 47](#_Toc121785749)

[Hình 19: Giao diện thêm sản phẩm 47](#_Toc121785750)

[Hình 20: Giao diện quản lí đơn hàng 48](#_Toc121785751)

[Hình 21: Giao diện trang cá nhân 48](#_Toc121785753)

[Hình 22: Giao diện lịch sử mua hàng 49](#_Toc121785754)

[Hình 23:Giao diện trang chủ 49](#_Toc121785755)

[Hình 24: Giao diện đánh giá 50](#_Toc121785756)

[Hình 25: Giao diện chính sách 50](#_Toc121785757)

[Hình 26: Giao diện vỏ hàng trống 51](#_Toc121785758)

[Hình 27: Giao diện vỏ hàng có sản phẩm 51](#_Toc121785759)

[Hình 28: Giao diện thanh toán 52](#_Toc121785760)

[Hình 29: Giao diện phần giới thiệu 52](#_Toc121785761)

Danh mục bảng biểu

[5.1. Ma trận Actor 16](#_Toc121806185)

[6.1.1. Use case tìm kiếm 18](#_Toc121806193)

[6.1.2. Use case thêm vào vỏ hàng 19](#_Toc121806194)

[6.1.3. Use case đặt hàng 20](#_Toc121806195)

[6.1.4. Use case xem danh sách sản phẩm 22](#_Toc121806196)

[6.1.5. Use case hình thức thanh toán 22](#_Toc121806197)

[6.1.6. Use case xem chi tiết sản phẩm 24](#_Toc121806198)

[6.1.7. Use case duyệt đơn hàng 25](#_Toc121806199)

[6.1.8. Use case thêm nhân viên 27](#_Toc121806200)

[6.1.9. Use case quản lý khách hàng 28](#_Toc121806201)

[6.1.10. Use case quản lý đơn hàng 30](#_Toc121806202)

[6.1.11. Use case sửa thông tin sản phẩm 31](#_Toc121806203)

[6.1.12. Use case xóa sản phẩm 32](#_Toc121806204)

[6.1.13. Use case đánh giá và bình luận 33](#_Toc121806205)

[6.1.14. Use case lọc sản phẩm 35](#_Toc121806206)

[6.1.15. Use case trạng thái vận chuyển 35](#_Toc121806207)

Bảng phân công công việc

|  |  |  |
| --- | --- | --- |
| Tên công việc | Mai Thị Quỳnh Như | Nguyễn Thị Thanh Ngân |
| Báo cáo | X | X |
| Thiết kế giao diện | X | X |
| Code | X | X |
| Up Host |  | X |

1. Giới thiệu chung
2. Giới thiệu nguyên nhân có hệ thống

* Cuộc sống, công việc ngày càng trở nên bận rộn vội vã. Từ đó mà nhu cầu mua hàng online cũng tăng cao. Hiện nay các sản phẩm hay dịch vụ đều có thể cung ứng trên thị trường online, có thể kể đến như quần áo, đồ ăn, thiết bị điện tử,... luôn được mọi người săn đơn. Tuy nhiên chúng ta không nên bỏ qua những thứ mang lại giá trị lịch sử ví dụ như tiền xưa mà ông bà trước đây đã đùng.
* Với xã hội hiện đại hiện nay thì tiền là vật không thể thiếu được để duy trì cuộc sống hằng ngày. Trải qua quá trình hình thành và phát triển, từ những đồng xu đầu tiên đến những loại tiền giấy cotton hay cao hơn là loại tiền Polymer đều có những chuyển biến rõ rệt.

1. Các khó khăn khi không có hệ thông

* Việc bắt tay kinh doanh online bằng các tài khoảng mạng xã hội như facebook, instagram thì vấn đề mua bán diễn ra rất lâu, vì quá trình tất cả quá trình nhận đơn hàng đều thực hiện bằng thủ công. Hay thử sức kinh doanh trên các sàn tương mại diện tử như shopee, tiki, lazada... ở đây việc kiểm duyệt sản phẩm lâu gây chậm trễ, gián đoạn quá trình kinh doanh hơn nữa sẽ khó xây dựng được thương hiệu của shop.

1. Các bất cập của hệ thống

* Sau một thời gian dài tìm hiểu hệ thống phần mềm quản lý, và cũng gặp phải nhiều bất cập:
* Hệ thống hiện có trở nên khá cứng nhắc, khi mà các quy trình trong hệ thống thường xuyên thay đổi theo các yêu cầu của thị trường và theo các yêu cầu quản lý.
* Chi phí vận hành cho hệ thống ngày càng trở nên lớn hơn.

1. Lợi ích của hệ thống mang lại

* Để tránh sự chậm trễ về việc cập nhập sản phẩm mới hay xây đựng thành công thương hiệu cho riêng shop chính là thiết kế website bán hàng riêng. Việc xây dựng một trang web online không chỉ giúp chủ shop tiết kiệm được một số chi phí mà website còn là một phần mở rộng thị trường và đa dạng hóa doanh thu. Ngoài ra chủ shop có thể tự viết những blog liên quan đến tiền xưa .

1. Yêu cầu chức năng

* Quản Lý: Quản lý người dùng, quản lý các nhân viên, cấp quyền sử dụng, đăng nhập hệ thống, quản lý thống kê.

+ Chức năng quản lý khách hàng:Xem danh sách khách hàng, xem thông tin khách hàng, quản lý khách đặt hàng.

+ Chức năng quản lý thống kê: xem xét thống kê doanh thu theo tháng.

* Nhân viên

+ Quản lý sản phẩm: Quản lý sản phẩm, xem danh sách sản phẩm, cập nhập sản phẩm.

+ Chức năng quản lý đơn hàng: Duyệt đơn hàng

1. Yêu cầu phi chức năng

* Người sử dụng: được cấp username và password và được phân quyền sử dụng web, giúp người quản lý có thể theo dõi, kiểm soát được nhân viên, tránh các hợp sửa đổi thông tin không thuộc phạm vi quyền hạn. Password của người sử dụng được mã hóa trước khi ghi vào dữ liệu và người sử dụng có thể đổi password của mình

1. Phân tích thiết kế hệ thống
2. Đặc tả hệ thống (System requirements)

“Website vuatien” là kênh bán hàng trực tuyến của cửa hàng cũng như thể hiện việc chăm sóc khách hàng ở mọi lúc, mọi nơi. Với website này khách hàng không cần đến các cửa hàng hay nhà phân phối mà vẫn tham khảo được tác dụng, giá cả và thông tin về từng sản phẩm sữa và sự đảm bảo từ phía nhà cung cấp. Tạo sự thuận tiện, thoải mái cho khách hàng khi mua sản phẩm.

Website vua tiền luôn hoạt động 24/24h có thể đón khách hàng vào bất cứ lúc nào. Dựa vào website, khách hàng có thể biết được chi tiết liên quan đến sản phẩm của web, Giá bao nhiêu?….

website bán hàng còn nhận được phản hồi của khách hàng một cách nhanh chóng và hiệu quả. Từ mục đánh giá của khách hàng mà nâng cao chất lượng sản phẩm cũng như địch vụ chăm sóc khách hàng. Mọi thắc mắc lớn khách hàng có thể liên hệ trực tiếp cho hai quản lý chính của hệ thống thông tin liên hệ tại mục giới thiệu. Chính sách của sản phẩm mà website vuatien luôn rõ ràng và minh bạch tạo thiện cảm cũng như sự tin tưởng cho khách hàng khi mua hàng.

Giao diện đẹp, thân thiện phù hợp với yêu cầu khách hàng.

Dễ thao tác quản lý.

Dễ sử dụng với các đối tượng người dùng, thuận tiện trong quản trị, dễ bảo trì.

Thông tin hiển thị chi tiết.

Chạy ổn định trên các trình duyệt

An toàn bảo mật thông tin người dùng.

1. Giả định (Assumption) cho hệ thống

* Thương mại điện tử giúp người bán
* Website bán hàng trực tiếp tới khách hàng với quy mô rộng, tốc độ nhanh và chi phí giảm rất nhiều so với các kênh bán hàng truyền thống khác
* Tại cùng 1 thời điểm, Website Thương mại điện tử có thể phục vụ hàng triệu lượtngười mua hàng ở khắp nơi trên thế giới với các yêu cầu rất khác nhau về thông tin sản phẩm, chủng loại sản phẩm, giá cả, hình ảnh, chất lượng, mẫu mã,...
* Thông tin, giá cả sản phẩm được cập nhật, thay đổi một cách tức thời theo chủ website. Hình ảnh sản phẩm chất lượng . Sản phẩm bố trí, trưng bày hấp dẫn, cuốn hút người xem, dẫn dắt khách hàng đi từ click này đến click khác. Sắp xếp sản phẩm trong chuyên mục theo nhiều loại như sản phẩm mới, sản phẩm nổi bật, sản phẩm bán chạy… để cho khách hàng dễ dàng lựa chọn, điền thông tin giá bán rõ ràng. Thiết lập kích thước ảnh: Hiển thị nhiều ảnh mô tả cho sản phẩm. Xem ảnh sản phẩm dạng slide show đẹp mắt , đánh giá, bình luận sản phẩm, thông tin giá: Giá bán
* Tính năng quản lý tồn kho. Tồn kho đóng vai trò rất quan trọng cho web kinh doanh online lẫn offline. Dư thừa quá nhiều hàng sẽ khiến việc kinh doanh chôn vốn, giảm hiệu quả kinh doanh còn thiếu hàng tồn sẽ gián đoạn cung cầu, dẫn đến việc mất khách. Mọi thay đổi, tăng giảm của hàng tồn sẽ được cập nhập kịp thời và nhanh chóng. Khi khách đặt đơn hàng thành công số lượng trong kho sẽ được cập nhập lại tương ứng.
* Tính năng giỏ hàng chuyên nghiệp, dễ dùng. Chức năng thêm sản phẩm vào giỏ hàng bằng một click chuột

+ Cập nhật thông tin giỏ hàng

+ Xóa sản phẩm khỏi giỏ hàng

+ Quy trình thanh toán và đặt hàng mới

* Tiếp nhận đơn đặt hàng trực tuyến=> Xem thông tin sản phẩm, giá trị đơn hàng => Xem thông tin người đặt hàng

+ Xác nhận đơn hàng thành công

+ Xác nhận hủy đơn hàng

+ Chức năng thanh toán online

Website đa dạng phương thức thanh toán:

+Khi mua hàng, ngày nay khách hàng có rất nhiều lựa chọn hình thức thanh toán khác nhau từ thanh toán tiền mặt COD cho đến thanh toán online. Vậy bạn đã đoán được, khi thiết kế website bán hàng cần những gì? đó chính là tính năng thanh toán.

+ Website bán hàng hiện nay phải tích hợp đa dạng hình thức thanh toán như COD, chuyển khoản, thẻ tín dụng, ví điện tử để đưa ra nhiều lựa chọn khác nhau cho khách hàng lựa chọn. Nắm bắt được xu thế đó website vuatien cũng thực hiện hình thức thanh toán khi đặt hàng tạo nhiều tiện lợi cho khách.

* Quản lý tài khoản khách hàng:Mọi thông tin khách hàng khi đăng kí tài khoản sẽ được cập nhập vào hệ thống web. Nhằm tránh tình trạnh đối thủ cạnh trang đặt nhiều đơn hàng rồi hủy thì web có chức năng nếu đặt 4 đơn liên tiếp mà không nhận hàng cả 4 đơn thì tài khoản sẽ bị khóa.
* Quản lý đơn hàng

+ Thông tin khách hàng khi đặt sản phẩm sẽ được hiện thị chi tiết như: họ và tên khách, số điện thoại của khách hàng, địa chỉ giao hàng, tổng tiền. Khi bấm vào chi tiết quản lý có thể xem khách đã đặt những gì, số lượng bao nhiêu, tất cả khoản phí rồi đến thành tiền. Việc quản lý đơn hàng còn giúp người quản lý web quản lý trình tự giao đơn hàng một cách nhanh chóng

+ Chức năng tìm kiếm : Chức năng này sẽ giúp khách hàng dễ dàng tìm được sản phẩm mà web đang mời chào: tìm kiếm theo từ khóa, danh mục, giá…. Ngoài ra mỗi danh mục sản phảm còn hiển thị ra số lượng sản phẩm hiện có. Điều này giúp chỉ wrb kiểm soát được số sản phẩm hiện có trên website của mình.

+ Có thể mua hàng thuận tiện, dễ dàng, nhanh chóng

+ Hình thức thanh toán mới tiện lợi, an toàn

+ Mở rộng sự chọn lựa khi mua hàng theo thị hiếu và nhu cầu

+ Có cơ hội mua sản phẩm trực tiếp từ nhà sản xuất hoặc nhà cung cấp chính không qua trung gian

1. Phạm vi của hệ thống (Scope)

Website giúp khách hàng được trải nghiệm dịch vụ tốt hơn, mua sắm tiện lợi hơn. Khi cần chỉ cần mất vài giây đăng nhập vào tài khoản để xem thông tin sản phẩm. Hệ thống tạo điều kiện thuận lợi cho việc lưu trữ, tìm kiếm thông tin một cách nhanh gọn.

Website bao gồm các loại người dùng chính: Khách hàng và quản lý:

+ Đối với khách hàng: Xem danh sách sản phẩm, xem chi tiết sản phẩm, xem số sản phẩm mà shop hiện có, đưa sản phảm vào giỏ hành, cập nhập giỏ hàng, tìm kiếm sản phẩm, đăng nhập, đắng kí, tùy chỉnh thông tinh…

+ Quản lý: là người nắm quyền điều hành trang web đương nhiên ngoài bao gồm những tính năng mà khách hàng có thể thực hiện thì quản lý có thể váo các chưc năng như quản lý sản phẩm, quản lý đơn hàng, thống kê doanh thu,…

1. Yêu cầu của người sử dụng hệ thống (Users requirements)

## Yêu cầu chức năng

- Quản lý: Quản lý người dùng

   + Chức năng quản lý khách hàng: Cập nhật danh sách khách hàng, xem thông tin khách hàng.

+ Quản lý sản phẩm: thêm, xóa, xữa sản phẩm.

   + Chức năng quản lý quá đơn hàng: xem danh sách đơn, cập nhập trạng thái đơn hàng.

-Nhân viên: Quản lý hợp đồng thuê xe, quản lý khách hàng

## Yêu cầu phi chức năng

Người sử dụng: được cấp username và password và được phân quyền sử dụng chương trình, giúp người quản lý có thể theo dõi, kiểm soát được chương trình, tránh các hợp sửa đổi thông tin không thuộc phạm vi quyền hạn.

Password của người sử dụng được mã hóa trước khi ghi vào dữ liệu và người sử dụng có thể đổi password của mình

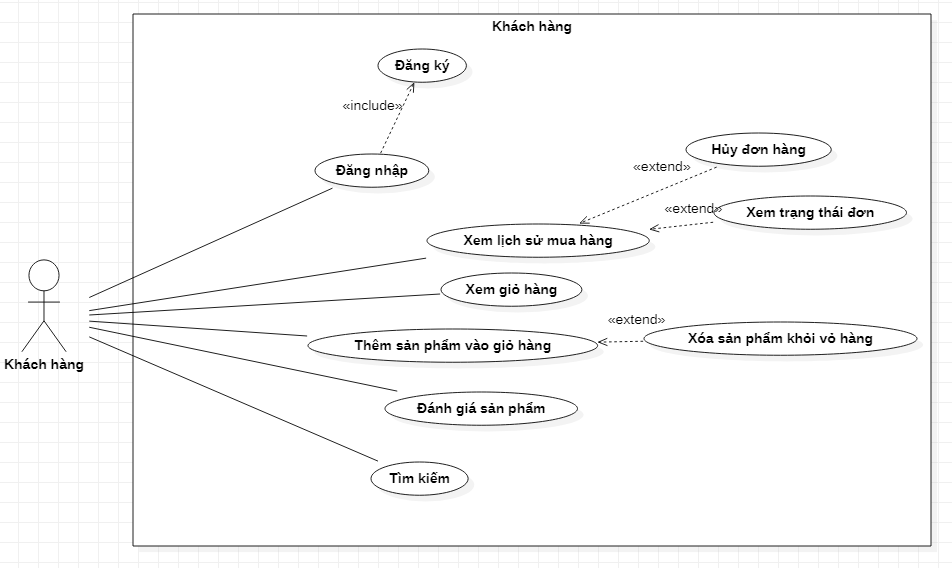
1. Ma trận actor và use case

### Ma trận Actor

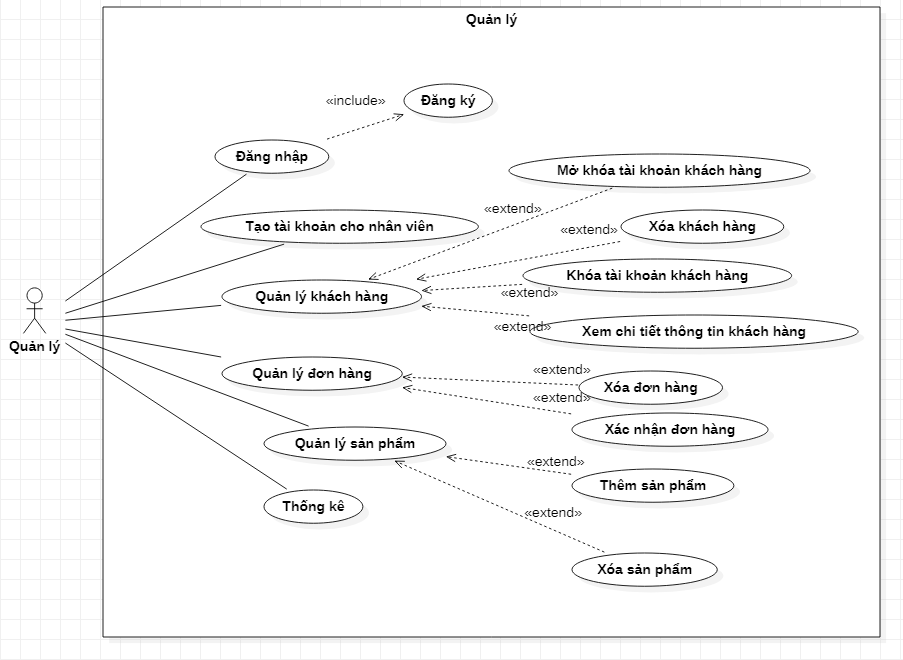
Table : Matrix actor

|  |  |  |
| --- | --- | --- |
| Actor  Use Case | User | Admin |
| UC01: Tìm kiếm | x | x |
| UC02: Vỏ hàng | x |  |
| UC03: Đặt hàng | x |  |
| UC04: Danh sách sản phẩm | x | x |
| UC05: Hình thức thanh toán | x | x |
| UC06: Xem chi tiết sản phẩm | x | x |
| UC07: Duyệt đơn |  | x |
| UC08: Thêm nhân viên |  | x |
| UC09: Quản lý khách hàng |  | x |
| UC10: Quản lý đơn hàng |  | x |
| UC11: Sửa thông tin sản phẩm |  | x |
| UC12: Xóa sản phẩm |  | x |
| UC13: Đánh giá và bình luận | x |  |
| UC14: Lọc sản phẩm | x | x |
| UC15: Trạng thái vận chuyển | x | x |

* 1. Use case



Hình 1: Usecase khách hàng



Hình 2: Usecase quản lí

1. Actor và use case

### Đặc tả use case

#### Use case tìm kiếm

­­Table : Use case tìm kiếm

|  |  |  |  |
| --- | --- | --- | --- |
| USName | Tìm kiếm | Code | UC01 |
| Description | Cho phép actoe dùng từ khóa để tìm kiếm trong hệ thống | | |
| Actor | Người dùng/ Quản lý | Trigger | Actor bấm vào khung tìm kiếm |
| Pre-condition |  | | |
| Post condition | Hiển thị sản phẩm dựa trên từ khóa tìm kiếm | | |

Activities

|  |  |  |  |
| --- | --- | --- | --- |
| Actor | | System | |
| Main Flow: Tìm kiếm | | | |
| 1 | Actor bấm vào khung tìm kiếm |  |  |
| 2 | Actor hàng nhập từ khoá tìm kiếm |  |  |
|  |  | 3 | Hệ thống truy cập vào bảng SanPham |
|  |  | 4 | Hệ thống tìm kiếm từ khóa gần đúng hoặc chính xác với từ khóa tìm kiếm |
|  |  | 5 | Hiển thị sản phẩm |
| Alternative Flow: | | | |
|  |  | 6 | Không tìm thấy sản phẩm chính xác với từ khóa |

Sytem Message

|  |  |
| --- | --- |
| MS01 | “Không có sản phẩm tìm kiếm”  Message thông báo không có sản phẩm như yêu cầu |

#### Use case thêm vào vỏ hàng

|  |  |  |  |
| --- | --- | --- | --- |
| USName | Thêm vào giỏ hàng | Code | UC02 |
| Description | Cho phép người dùng chọn sản phẩm yêu thích vào giỏ hàng | | |
| Actor | Người dùng | Trigger | Actor bấm nút đặt hàng kế bên sản phẩm |
| Pre-condition |  | | |
| Post condition | Hiển thị sản phẩm đã chọn | | |

Activities

|  |  |  |  |
| --- | --- | --- | --- |
| Actor | | System | |
| Main Flow: Thêm vào giỏ hàng | | | |
| 1 | Khách hàng bấm vào xem chi tiết sản phẩm |  |  |
| 2 | Khách hàng bấm vào thêm hàng vào giỏ |  |  |
|  |  | 3 | Hệ thống truy cập vào bảng SanPham |
|  |  | 4 | Hệ thống kiểm tra số lượng hàng còn không |
|  |  | 5 | Số lượng nhập vào hợp lệ, thêm vào giỏ hàng thành công |
| Alternative Flow: | | | |
|  |  | 6 | Không đủ số lượng |

Sytem Message

|  |  |
| --- | --- |
| MS02 | “Không còn đủ sản phẩm”  Message thông báo khi số lượng sản phẩm không đủ |

#### Use case đặt hàng

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Đặt hàng | Code | UC03 |
| Description | Người dùng vào giỏ hàng kiểm tra sản phẩm muốn đặt | | |
| Actor | Người dùng | Trigger | Actor chọn giỏ hàng rồi nhấn nút đặt hàng phía dưới cùng |
| Pre-condition |  | | |
| Post condition | Chuyển tới trang quá trình vận chuyển | | |

Activities

|  |  |  |  |
| --- | --- | --- | --- |
| Actor | | System | |
| Main Flow: Thêm vào giỏ hàng | | | |
| 1 | Khách hàng bấm vào cỏ hàng |  |  |
| 2 | Khách hàng bấm vào đặt hàng |  |  |
|  |  | 3 | Hệ thống truy cập vào bảng SanPham, KhachHang |
|  |  | 4 | Hệ thống tự động điền thông tin đặt hàng dựa trên bảng SanPham, KhachHang |
|  |  | 5 | Hệ thống yêu cầu khách hàng nhập địa chỉ nhận hàng và chọn hình thức thanh toán |
| 6 | Khách hàng nhập địa chỉ nhận hàng và chọn hình thức thanh toán |  |  |
| 7 | Khách hàng xác nhận đặt hàng |  |  |
|  |  | 8 | Hệ thống lưu thông tin và gửi thông báo xác nhận đến admin |

#### Use case xem danh sách sản phẩm

|  |  |  |  |
| --- | --- | --- | --- |
| USName | Xem danh sách sản phẩm | Code | UC04 |
| Description | Cho phép actor xem danh sách sản phẩm | | |
| Actor | Khách hàng, admin | Trigger | Actor bấm vào trang chủ |
| Pre-condition |  | | |
| Post condition | Hiển thị danh sách sản phẩm | | |

Activities

|  |  |  |  |
| --- | --- | --- | --- |
| **Actor** | | **System** | |
| **Main Flow: Xem danh sách sản phẩm** | | | |
| 1 | Actor bấm vào nút trang chủ |  |  |
|  |  | 2 | Hệ thống truy cập vào bảng SanPham |
|  |  | 3 | Load thông tin các sản phẩm lên màn hình |

#### Use case hình thức thanh toán

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Hình thức thanh toán online | Code | UC05 |
| Description | Người dùng chọn hình thức thanh toán khi đặt hàng sản phẩm | | |
| Actor | Người dùng | Trigger | Chọn hình thức thanh toán |
| Pre-condition |  | | |
| Post condition | Chuyển tới trang xác nhập thanh toán | | |

Activities

|  |  |  |  |
| --- | --- | --- | --- |
| Actor | | System | |
| Main Flow: Thanh toán online | | | |
| 1 | Khách hàng đăng nhập vào hệ thống |  |  |
| 2 | Khách hàng bấm vào giỏ hàng |  |  |
| 3 | Khách hàng bấm vào giỏ hàng |  |  |
|  |  | 4 | Hệ thống chuyển tới trang thông tin đặt hàng |
| 5 | Khách hàng nhập đầy đủ thông tin đặt hàng và chọn thanh toán băng momo |  |  |
| 6 | Khách hàng bấm vào giỏ hàng |  |  |
|  |  | 7 | Hệ thống gửi mã giao dịch đến momo |
|  |  | 8 | Momo gửi mã QR về hệ thống |
|  |  | 9 | Hệ thống hiển thị mã QR do momo cấp |
| 10 | Người dùng quét mã QR để thanh toán |  |  |
|  |  | 11 | Momo xác thực thanh toán, trừ tiền của khách hàng và gửi thông tin về hệ thống |
|  |  | 12 | Hệ thống hiển thị thanh toán thành công |
| Alternative Flow: | | | |
|  |  | 13 | Mã QR vô hiệu khi quá 10 phút, thanh toán thất bại trở về nhập thông tin đơn hàng |

|  |  |
| --- | --- |
| MS05 | “Thanh toán thất bại”  Message thông báo khi mã QR bị vô hiệu |

System Mesage

#### Use case xem chi tiết sản phẩm

|  |  |  |  |
| --- | --- | --- | --- |
| USName | Xem chi tiết sản phẩm | Code | UC06 |
| Description | Cho phép người xem thông tin cụ thể của sản phẩm | | |
| Actor | Người dùng | Trigger | Actor bấm vào sản phẩm muốn xem chi tiết |
| Pre-condition |  | | |
| Post condition | Hiển thị các thông tin liên quan đến sản phẩm muốn xem chi tiết | | |

Activities

|  |  |  |  |
| --- | --- | --- | --- |
| Actor | | System | |
| Main Flow: Xem danh sách sản phẩm | | | |
| 1 | Actor bấm vào sản phẩm muốn xem chi tiết. |  |  |
|  |  | 2 | Hệ thống truy cập vào bảng SanPham |
|  |  | 3 | Load thông tin chi tiết sản phẩm được chọn lên màn hình |

#### Use case duyệt đơn hàng

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Duyệt đơn hàng | Code | UC07 |
| Description | Hiển thị tình trang đơn hàng (trạng thái đơn hàng) | | |
| Actor | Nhân viên | Trigger | Actor chọn danh mục quản lý đơn hàng |
| Pre-condition |  | | |
| Post condition | Dẫn tới trang đơn hàng thể xem trạng thái | | |

Activites

|  |  |  |  |
| --- | --- | --- | --- |
| **Actor** | | **System** | |
| **Main Flow: Duyệt đơn hàng** | | | |
| 1 | Nhân viên đăng nhập vào hệ thống |  |  |
| 2 | Nhân viên nhấn vào mục quản lý đơn hàng |  |  |
|  |  | 3 | Hệ thống truy cập vào bảng HoaDon |
|  |  | 4 | Hiển thị danh sách đơn hàng |
| 5 | Bấm vào nút xác nhận tại đơn  hàng cần xác nhận |  |  |
|  |  | 6 | Hệ thống xác nhận đơn hàng |

|  |  |
| --- | --- |
| MS0701 | “Không có đơn hàng”  Message thông báo chưa có đơn hàng |
| MS0702 | “Đơn hàng bị hủy do khách hàng”  Message thông báo đơn hàng bị hủy do khách hàng |
| MS0703 | “Đơn hàng bị hủy do người bán”  Message thông báo đơn hàng bị hủy do người bán (lý do) |

System

#### Use case thêm nhân viên

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Thêm nhân viên | Code | UC08 |
| Description | Quản lý thêm thông tin nhân viên và tài khoản nhân viên trong hệ thống. | | |
| Actor | Quản lý | Trigger | Actor chọn danh mục nhân viên và thêm mới nhân viên |
| Pre-condition |  | | |
| Post condition | Nhân viên đã được tạo thành công có thể dùng đăng nhập vào hệ thống | | |

Activities

|  |  |  |  |
| --- | --- | --- | --- |
| **Actor** | | **System** | |
| **Main Flow: Thêm nhân viên** | | | |
| 1 | Quản lý đăng nhập vào hệ thống |  |  |
| 2 | Quản lý bấm vào mục quản lý nhân viên |  |  |
|  |  | 3 | Hệ thống truy cập vào bảng NguoiDung và lọc ra nhân viên |
|  |  | 4 | Hiển thị nhân viên ra màn hình |
| 5 | Quản lý bấm vào nút thêm nhân viên |  |  |
| 6 | Quản lý nhập đầy đủ thông tin nhân viên mới và bấm tạo mới |  |  |
|  |  | 7 | Hệ thống kiểm tra thông tin nhập vào |
|  |  | 8 | Thông tin hợp lệ, thêm nhân viên thành công |
| **Alternative Flow:** | | | |
|  |  | 9 | Thông tin không hợp lệ hoặc bị thiếu |

System Mesage

|  |  |
| --- | --- |
| MS08 | “Thông tin không hợp lệ”  Message thông báo khi thông tin nhập vào không hợp lệ hoặc bị thiếu |

#### Use case quản lý khách hàng

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Quản lý khách hàng | Code | UC09 |
| Description | Nhân viên quản lý tài khoản của khách hàng có quyền thêm, xóa, sửa khách hàng | | |
| Actor | Quản lý | Trigger | Actor chọn danh mục quản lý khách hàng để xem các thông tim liên quan |
| Pre-condition |  | | |
| Post condition | Khi nhân viên thực hiện thao tác và ấn lưu thì khách hàng sẽ thấy những thay đổi đó trên phần tài khoản của mình. | | |

Activities

|  |  |  |  |
| --- | --- | --- | --- |
| **Actor** | | **System** | |
| **Main Flow: Quản lý khách hàng** | | | |
| 1 | Quản lý đăng nhập vào hệ thống |  |  |
| 2 | Quản lý bấm vào mục quản lý khách hàng |  |  |
|  |  | 3 | Hệ thống truy cập vào bảng tài khoản và hiện thị danh sách tài khoản có trong database |
|  |  | 4 | Hiển thị khách hàng ra màn hình |
| 5 | Nhân viên bấm vào nút thêm, xóa, sửa khách hàng tùy theo thao tác cần thực hiện |  |  |
|  |  | 6 | Hệ thống kiểm tra thông tin |
|  |  | 7 | Thông tin hợp lệ, thì lưu lại thao tác vừa rồi. |
| **Alternative Flow:** | | | |
|  |  | 8 | Thông tin không hợp lệ hoặc bị thiếu |

System Mesage

|  |  |
| --- | --- |
| MS08 | “Thông tin không hợp lệ”  Message thông báo khi thông tin nhập vào không hợp lệ hoặc bị thiếu |

#### Use case quản lý đơn hàng

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Quản lý đơn hàng | Code | UC10 |
| Description | Nhân viên quản lý đơn hàng của khách hàng chuyển trạng thái đơn hàng, xác nhận đơn hàng | | |
| Actor | Quản lý | Trigger | Actor chọn danh mục quản lý đơn hàng để xem các thông tim liên quan |
| Pre-condition |  | | |
| Post condition | Khi nhân viên thực hiện thao tác đổi trạng thái đơn hàng thì trạng thái đơn hàng của tài khoản khách hàng cũng được cập nhật theo. | | |

Activities

|  |  |  |  |
| --- | --- | --- | --- |
| **Actor** | | **System** | |
| **Main Flow: Quản lý đơn hàng** | | | |
| 1 | Nhân viên đăng nhập vào hệ thống |  |  |
| 2 | Nhân viên nhấn vào mục quản lý đơn hàng |  |  |
|  |  | 3 | Hệ thống truy cập vào bảng HoaDon |
|  |  | 4 | Hiển thị danh sách đơn hàng |
| 5 | Bấm vào nút hoàn thành đơn hàng để xác nhận giao hàng thành công |  |  |
|  |  | 6 | Hệ thống chuyển sang đơn hàng thành công |

#### Use case sửa thông tin sản phẩm

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Sửa sản phẩm | Code | UC11 |
| Description | Nhân viên cập nhập lại sản phẩm | | |
| Actor | Nhân viên | Trigger | Actor chọn danh mục sản phẩm cần sửa lại |
| Pre-condition |  | | |
| Post condition | Sản phẩm sửa thành công sẽ hiển thị ngay trên web | | |

Activities

|  |  |  |  |
| --- | --- | --- | --- |
| **Actor** | | **System** | |
| **Main Flow: Sửa thông tin sản phẩm** | | | |
| 1 | Nhân viên đăng nhập vào hệ thống |  |  |
| 2 | Nhân viên nhấn vào mục quản lý sản phẩm |  |  |
|  |  | 3 | Hệ thống truy cập vào bảng sản phẩm |
|  |  | 4 | Hiển thị danh sách sản phẩm |
| 5 | Chọn sản phẩm cần sửa thông tin |  |  |
| 6 | Sửa thông tin cần sửa và ấn nút lưu thông tin |  |  |
|  |  | 6 | Hệ thống cập nhật lại thông tin nhân viên đã sửa và lưu và hệ thống |

#### Use case xóa sản phẩm

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Xóa sản phẩm | Code | UC12 |
| Description | Quản lý cập nhập lại sản phẩm | | |
| Actor | Quản lý | Trigger | Actor chọn danh mục sản phẩm cần xóa |
| Pre-condition |  | | |
| Post condition | Sản phẩm xóa thành công sẽ hiển thị ngay trên web | | |

Activities

|  |  |  |  |
| --- | --- | --- | --- |
| **Actor** | | **System** | |
| **Main Flow: Sửa thông tin sản phẩm** | | | |
| 1 | Nhân viên đăng nhập vào hệ thống |  |  |
| 2 | Nhân viên nhấn vào mục quản lý sản phẩm |  |  |
|  |  | 3 | Hệ thống truy cập vào bảng sản phẩm |
|  |  | 4 | Hiển thị danh sách sản phẩm |
| 5 | Chọn sản phẩm cần xóa |  |  |
| 6 | Ấn nút xóa thông tin |  |  |
|  |  | 6 | Hệ thống cập nhật lại thông tin nhân viên đã xóa và xóa khỏi hệ thống |

#### Use case đánh giá và bình luận

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Đánh giá và bình luận | Code | UC13 |
| Description | Khách hàng gửi bình luận về sản phẩm mình đã mua | | |
| Actor | Khách hàng | Trigger | Actor chọn mục bình luận |
| Pre-condition |  | | |
| Post condition | Khách hàng gửi bình luận thì sẽ được lưu trong hệ thống và hiện lên tron phần xem chi tiết sản phẩm | | |

Activities

|  |  |  |  |
| --- | --- | --- | --- |
| Actor | | System | |
| Main Flow: Đánh giá và bình luận sản phẩm | | | |
| 1 | Khách hàng đăng nhập vào hệ thống |  |  |
| 2 | Khách hàng bấm vào mục lịch sử mua hàng |  |  |
|  |  | 3 | Hệ thống truy xuất vào bảng HoaDon |
|  |  | 4 | Hệ thống xuất hóa đơn ra màn hình |
| 5 | Khách hàng bấm vào tab đã giao |  |  |
| 6 | Khách hàng bấm vào hóa đơn |  |  |
|  |  | 7 | Hệ thống hiển thị chi tiết hóa đơn |
| 8 | Khách hàng bấm vào nút bình luận và đánh giá |  |  |
| 9 | Khách hàng nhập bình luận và đánh giá của mình và bấm nút đánh giá |  |  |
|  |  | 10 | Hệ thống xác nhận đánh giá của khách hàng và chuyển đến cho quản lý |
|  |  | 11 | Quản lý hoặc nhân viên xác nhận thì hệ thống hiển thị đánh giá của khách hàng vào sản phẩm |

#### Use case lọc sản phẩm

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Lọc sản phẩm | Code | UC14 |
| Description | Người vào danh mục sản phẩm | | |
| Actor | Người dùng | Trigger | Chọn lọc theo giá tiền hoặc đánh giá |
| Pre-condition |  | | |
| Post condition | Hiển thị lại sản phẩm đã qua bộ lọc | | |

#### Use case trạng thái vận chuyển

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Trạng thái vận chuyển | Code | UC15 |
| Description | Hiển thị chi tiết vận chuyển đơn hàng (trạng thái đơn hàng) | | |
| Actor | Quản lý/ Khách hàng | Trigger | Actor chọn danh mục đơn hàng |
| Pre-condition |  | | |
| Post condition | Dẫn tới trang vận chuyển đơn hàng thể xem trạng thái | | |

### Activity diagram

#### Diagram Description automatically generatedActivity Diagram Tìm kiếm

Hình 3: Activity Diagram tìm kiếm

#### Diagram Description automatically generatedActivity Diagram Thêm vào giỏ hàng

Hình 4: Activity Diagram Thêm giỏ hàng

#### Xem giỏ hàng

Diagram

Description automatically generated

Hình 5: Activity Diagram xem giỏ hàng

#### Diagram Description automatically generatedActivity Diagram Đặt Hàng

Hình 6: Activity Diagram đặt hàng

#### Activity Diagram Hình thức thanh toán

Diagram, schematic

Description automatically generated

Hình 7: Activity Diagram thanh toán

#### Đánh giá và bình luận sản phẩm

Diagram

Description automatically generated

Hình 8: Activity Diagram bình luận sản phẩm

#### Duyệt đơn hàng

Diagram

Description automatically generated

Hình 9: Activity Diagram duyệt đơn hàng

#### Activity Diagram Xóa sản phẩm

A picture containing chart

Description automatically generated

Hình 10: Activity Diagram xóa sản phẩm

#### Diagram, schematic Description automatically generatedActivity Diagram Lọc sản phẩm

Hình 11: Activity Diagram lọc sản phẩm

#### Activity Diagram Xem chi tiết sản phẩm

Diagram

Description automatically generated

Hình 12: xem chi tiết sản phẩm

#### Activity Diagram Thêm sản phẩm

Diagram

Description automatically generated

Hình 13: Activity Diagram Thêm sảm phẩm

#### Activity Diagram Sửa sản phẩm

Diagram

Description automatically generated

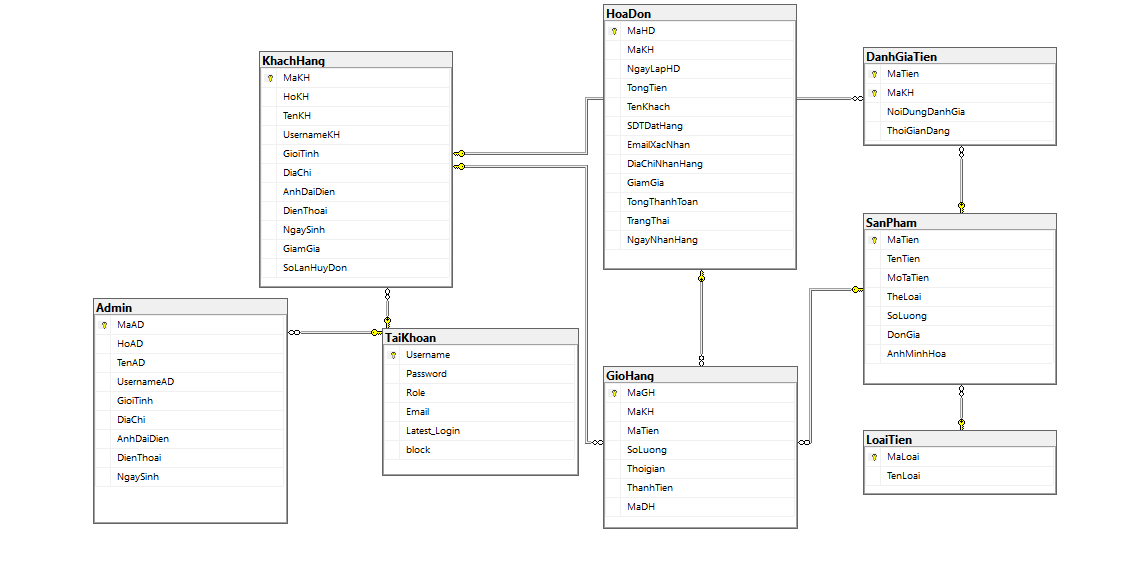
Hình 14: Activity Diagram sửa sản phẩm

#### Thêm nhân viên

Diagram

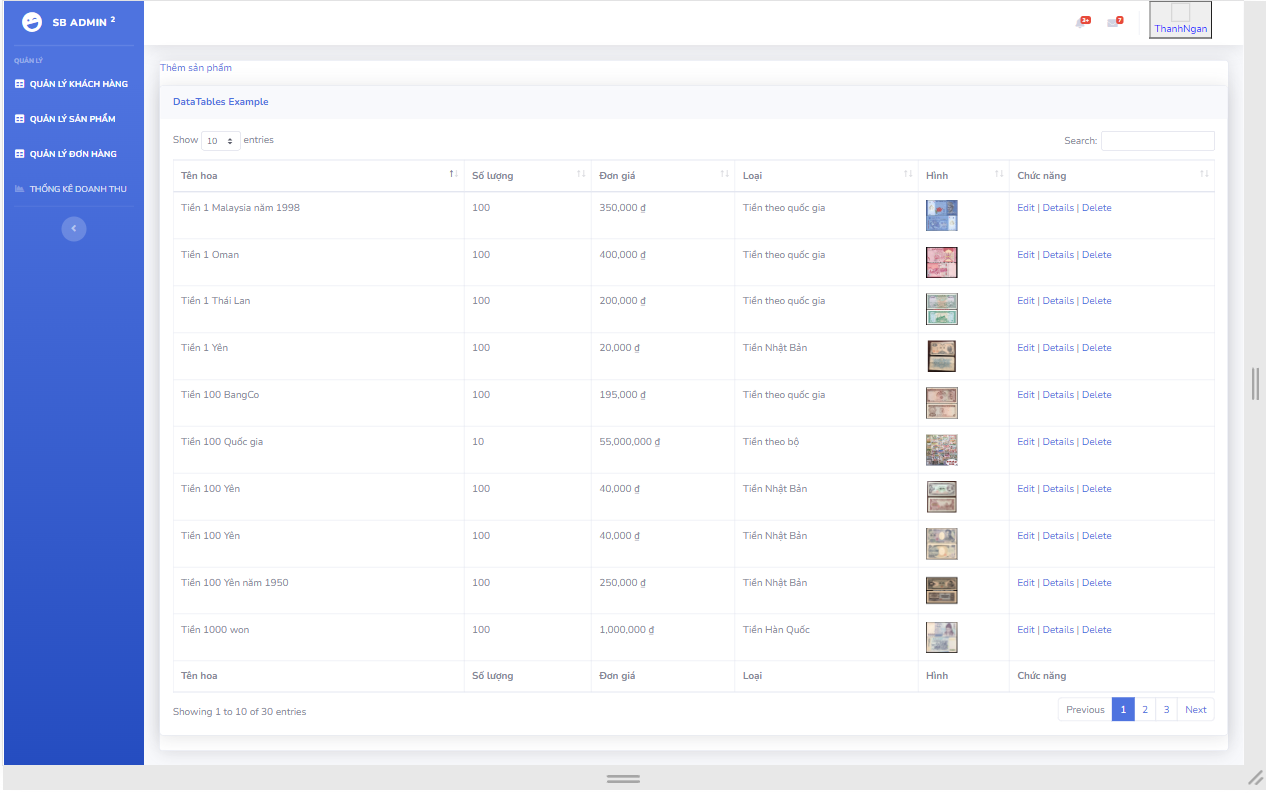
Description automatically generated

Hình 15: Activity Diagram thêm nhân viên

1. Sơ đồ ERD

Hình 16: ERD

1. Giao diện
   1. Giao diện admin



Hình 18: Giao diện quản lý sản phẩm

Graphical user interface, text, application, email

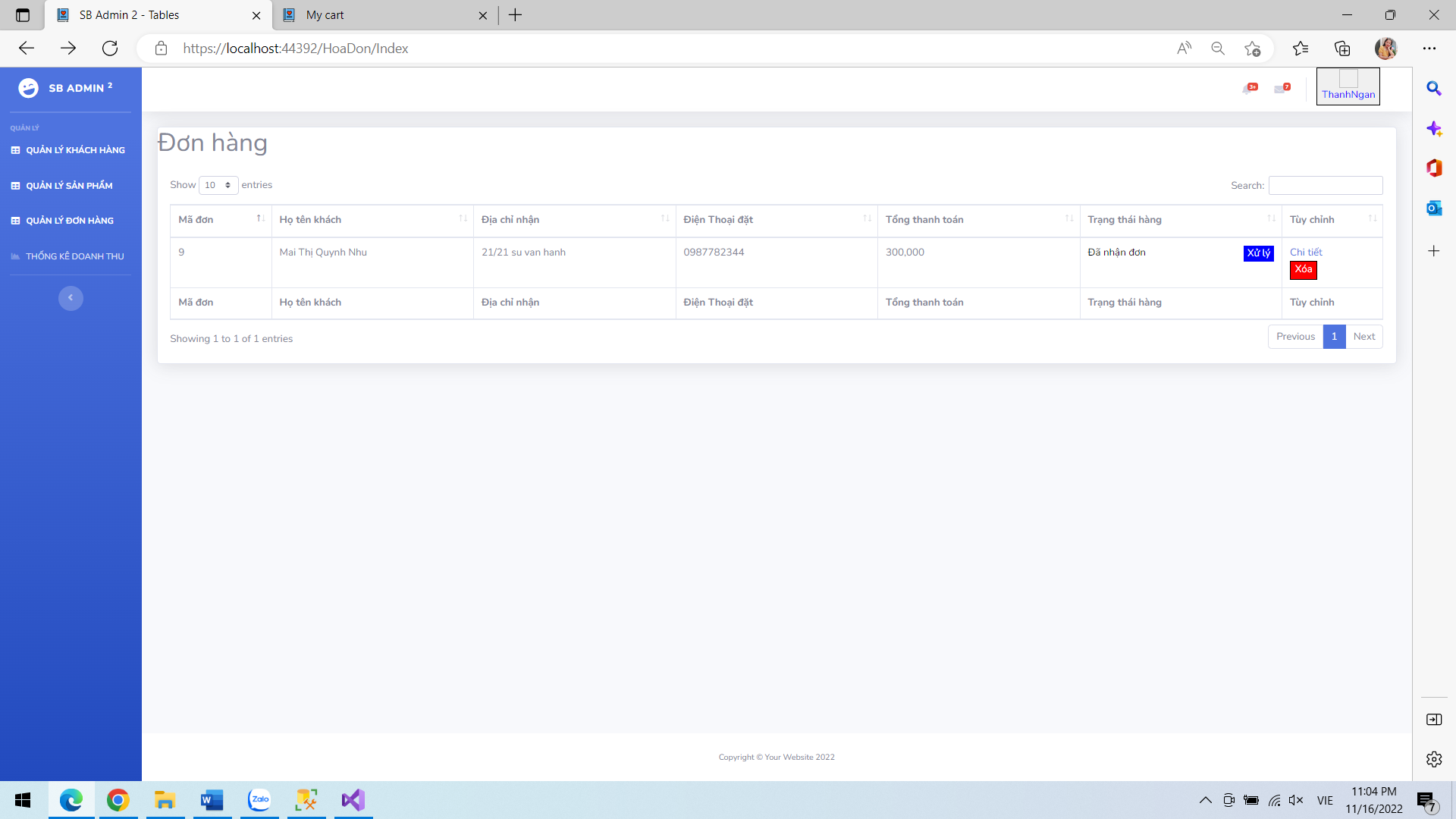
Description automatically generated

Hình 18: Giao diện quản lí khách hàng

Graphical user interface, application, table, Excel

Description automatically generated

Hình 19: Giao diện thêm sản phẩm



Hình 20: Giao diện quản lí đơn hàng

### Giao diện user

Graphical user interface, text

Description automatically generated

Hình 21: Giao diện trang cá nhân

Graphical user interface, text

Description automatically generated

Hình 22: Giao diện lịch sử mua hàng

Graphical user interface, website

Description automatically generated

Hình 23:Giao diện trang chủ

Graphical user interface, application

Description automatically generated

Hình 24: Giao diện đánh giá

Graphical user interface, text, application, email

Description automatically generated

Hình 25: Giao diện chính sách

Graphical user interface, application

Description automatically generated

Hình 26: Giao diện vỏ hàng trống

Graphical user interface, text, application

Description automatically generated

Hình 27: Giao diện vỏ hàng có sản phẩm

Graphical user interface, text, application

Description automatically generated

Hình 28: Giao diện thanh toán

Graphical user interface, application

Description automatically generated

Hình 29: Giao diện phần giới thiệu

1. Code
   1. Code admin

#### HoaDonController

using System;

using System.Collections.Generic;

using System.Data.Entity;

using System.Data.Entity.Validation;

using System.Linq;

using System.Net;

using System.Net.Mail;

using System.Web;

using System.Web.Mvc;

using BanTien.Models;

namespace BanTien.Controllers

{

public class HoaDonController : Controller

{

BANTIENEntities db = new BANTIENEntities();

/\* Detailed receipt \*/

[Authorize]

[HttpGet]

public ActionResult Index()

{

var hoadon = db.HoaDons.ToList();

return View(hoadon);

}

/\* Detailed receipt \*/

[Authorize]

[HttpGet]

public ActionResult Delete(int? id)

{

if (id == null)

return new HttpStatusCodeResult(HttpStatusCode.BadRequest);

return RedirectToAction("Index");

}

/\* Detailed receipt \*/

[HttpGet]

[Authorize]

public ActionResult Invoice(int? id)

{

if (id == null)

return new HttpStatusCodeResult(HttpStatusCode.BadRequest);

var hoadon = (from u in db.HoaDons

where u.MaHD == id

select u).FirstOrDefault();

InvoiceDetail inv = new InvoiceDetail();

inv.MaDH = id;

inv.MaKH = hoadon.MaKH;

inv.Ho\_TenKH = hoadon.TenKhach;

inv.SDTDatHang = hoadon.SDTDatHang;

inv.EmailXacNhan = hoadon.EmailXacNhan;

inv.DiaChiNhanHang = hoadon.DiaChiNhanHang;

inv.NgayLapHD = hoadon.NgayLapHD;

inv.NgayNhan = hoadon.NgayNhanHang;

inv.TrangThai = hoadon.TrangThai;

inv.SachDaMua = new List<ItemPayment>();

var giohang = (from u in db.GioHangs

where u.MaDH == id

select u).ToList();

for (int i = 0; i < giohang.Count; i++)

{

ItemPayment item = new ItemPayment();

item.STT = i + 1;

item.TenTien = giohang[i].SanPham.TenTien;

item.SoLuong = giohang[i].SoLuong;

item.DonGia = giohang[i].SanPham.DonGia;

item.ThanhTien = giohang[i].ThanhTien;

inv.SachDaMua.Add(item);

}

inv.GiamGia = hoadon.GiamGia;

inv.TongTien = hoadon.TongTien;

inv.TongThanhToan = hoadon.TongThanhToan;

return View(inv);

}

/\* Cancel \*/

[HttpPost]

[Authorize]

public ActionResult Cancel(int idDon, int idKhach)

{

var giohang = (from u in db.GioHangs

where u.MaDH == idDon

select u).ToList();

var hoadon = db.HoaDons.Find(idDon);

// Hoàn lại toàn bộ số lượng vào kho và xóa giỏ

foreach(var item in giohang)

{

item.SanPham.SoLuong += item.SoLuong;

db.Entry(item.SanPham).State = EntityState.Modified;

db.GioHangs.Remove(item);

}

db.HoaDons.Remove(hoadon);

var khach = db.KhachHangs.Find(idKhach);

khach.SoLanHuyDon ++;

if(khach.SoLanHuyDon >= 4)

{

var block\_user = db.TaiKhoans.Find(khach.UsernameKH);

block\_user.block = true;

db.Entry(block\_user).State = EntityState.Modified;

}

db.Entry(khach).State = EntityState.Modified;

db.SaveChanges();

return RedirectToAction("Index", "Home");

}

/\* Get shopping history \*/

[HttpGet]

[Authorize]

public ActionResult PaymentHistory(int? id)

{

if(id == null)

return new HttpStatusCodeResult(HttpStatusCode.BadRequest);

var hoadon = (from u in db.HoaDons

where u.MaKH == id

select u).ToList();

return View(hoadon);

}

/\* Create new payment \*/

[HttpPost]

[Authorize]

public ActionResult Create(CreateReceipt model)

{

var giohang = (from u in db.GioHangs

where u.MaKH == model.MaKH && u.MaDH == null

select u).ToList();

HoaDon hd = new HoaDon();

hd.MaKH = model.MaKH;

hd.TenKhach = model.Ho\_TenKH;

hd.NgayLapHD = DateTime.Now;

hd.SDTDatHang = model.SDTDatHang;

hd.EmailXacNhan = model.EmailXacNhan;

hd.DiaChiNhanHang = model.DiaChiNhanHang;

hd.GiamGia = model.GiamGia;

hd.TongTien = model.TongTien;

hd.TongThanhToan = model.TongThanhToan;

hd.TrangThai = "Đã nhận đơn";

hd.NgayNhanHang = DateTime.Today.AddDays(5);

db.HoaDons.Add(hd);

db.SaveChanges();

foreach(var item in giohang)

{

item.MaDH = hd.MaHD;

db.Entry(item).State = EntityState.Modified;

}

db.SaveChanges();

//string sms = "<html><body><table border='1'> <caption>Thông tin đơn</caption><tr><th>Mã đơn</th> <th>Ngày lập </th><th> Ngày nhận</th><th> Trạng thái</th><th>Tổng thanh toán</th> </tr>";

//sms += "<tr>";

//sms += "<td>" + hd.MaHD + "</td>";

//sms += "<td>" + hd.NgayLapHD + "</td>";

//sms += "<td>" + hd.NgayNhanHang + "</td>";

//sms += "<td>" + hd.TrangThai + "</td>";

//sms += "<td>" + hd.TongThanhToan + "</td>";

//sms += "</tr>";

//sms += "</table>";

//// Gửi mail

//MailMessage mail = new MailMessage("ducphung51194@gmail.com", hd.EmailXacNhan, "Thông tin đơn hàng", sms);

//SmtpClient client = new SmtpClient("smtp.gmail.com", 587);

//client.EnableSsl = true;

//client.Credentials = new NetworkCredential("ducphung51194@gmail.com", "vinhduc147a");

//mail.IsBodyHtml = true;

//client.Send(mail);

return RedirectToAction("Index", "Home");

}

[Authorize]

public ActionResult NextProcess(int? id, int type)

{

var donhang = db.HoaDons.Find(id);

if(type == 1)

donhang.TrangThai = "Đang xử lý";

else

if(type == 2)

donhang.TrangThai = "Đã giao hàng";

db.Entry(donhang).State = EntityState.Modified;

db.SaveChanges();

return RedirectToAction("Index", "HoaDon");

}

[Authorize]

public ActionResult AdminDelete(int? id)

{

var donhang = db.HoaDons.Find(id);

db.HoaDons.Remove(donhang);

db.SaveChanges();

return RedirectToAction("Index", "HoaDon");

}

}

}

#### KhachHangsController

using System;

using System.Collections.Generic;

using System.Data;

using System.Data.Entity;

using System.Linq;

using System.Net;

using System.Web;

using System.Web.Mvc;

using BanTien.Models;

namespace BanTien.Controllers

{

public class KhachHangsController : Controller

{

private BANTIENEntities db = new BANTIENEntities();

// GET: KhachHangs

[Authorize]

public ActionResult Index(string searchString)

{

var listKH = db.KhachHangs.ToList();

if (!String.IsNullOrEmpty(searchString))

{

var links = db.KhachHangs.Where(s => s.UsernameKH.Contains(searchString)

|| s.TenKH.Contains(searchString));

return View(links.ToList());

}

else

return View(listKH);

}

// GET: KhachHangs/Details/5

public ActionResult Details(int? id)

{

if (id == null)

{

return new HttpStatusCodeResult(HttpStatusCode.BadRequest);

}

var khachHang = db.KhachHangs.Find(id);

if (khachHang == null)

{

return HttpNotFound();

}

return View(khachHang);

}

// GET: KhachHangs/Create

public ActionResult Create()

{

ViewBag.UsernameKH = new SelectList(db.TaiKhoans, "Username", "Password");

return View();

}

// POST: KhachHangs/Create

// To protect from overposting attacks, enable the specific properties you want to bind to, for

// more details see https://go.microsoft.com/fwlink/?LinkId=317598.

[HttpPost]

[ValidateAntiForgeryToken]

public ActionResult Create([Bind(Include = "MaKH,HoKH,TenKH,UsernameKH,GioiTinh,DiaChi,AnhDaiDien,DienThoai,NgaySinh,GiamGia")] KhachHang khachHang)

{

if (ModelState.IsValid)

{

db.KhachHangs.Add(khachHang);

db.SaveChanges();

return RedirectToAction("Index");

}

ViewBag.UsernameKH = new SelectList(db.TaiKhoans, "Username", "Password", khachHang.UsernameKH);

return View(khachHang);

}

// GET: KhachHangs/Edit/5

public ActionResult Edit(int? id)

{

if (id == null)

{

return new HttpStatusCodeResult(HttpStatusCode.BadRequest);

}

KhachHang khachHang = db.KhachHangs.Find(id);

if (khachHang == null)

{

return HttpNotFound();

}

ViewBag.UsernameKH = new SelectList(db.TaiKhoans, "Username", "Password", khachHang.UsernameKH);

return View(khachHang);

}

// POST: KhachHangs/Edit/5

// To protect from overposting attacks, enable the specific properties you want to bind to, for

// more details see https://go.microsoft.com/fwlink/?LinkId=317598.

[HttpPost]

public ActionResult Edit([Bind(Include = "MaKH,HoKH,TenKH,UsernameKH,GioiTinh,DiaChi,AnhDaiDien,DienThoai,NgaySinh,GiamGia")] KhachHang khachHang)

{

if (ModelState.IsValid)

{

db.Entry(khachHang).State = EntityState.Modified;

db.SaveChanges();

return RedirectToAction("Index");

}

ViewBag.UsernameKH = new SelectList(db.TaiKhoans, "Username", "Password", khachHang.UsernameKH);

return View(khachHang);

}

// GET: KhachHangs/Delete/5

public ActionResult Delete(int? id)

{

if (id == null)

{

return new HttpStatusCodeResult(HttpStatusCode.BadRequest);

}

KhachHang khachHang = db.KhachHangs.Find(id);

if (khachHang == null)

{

return HttpNotFound();

}

return View(khachHang);

}

// POST: KhachHangs/Delete/5

[HttpPost, ActionName("Delete")]

[ValidateAntiForgeryToken]

public ActionResult DeleteConfirmed(int id)

{

KhachHang khachHang = db.KhachHangs.Find(id);

db.KhachHangs.Remove(khachHang);

db.SaveChanges();

return RedirectToAction("Index");

}

protected override void Dispose(bool disposing)

{

if (disposing)

{

db.Dispose();

}

base.Dispose(disposing);

}

}

}

#### SanPhamsController

using System;

using System.Collections.Generic;

using System.Data;

using System.Data.Entity;

using System.Linq;

using System.Net;

using System.Web;

using System.Web.Mvc;

using BanTien.Models;

namespace BanTien.Controllers

{

public class SanPhamsController : Controller

{

private BANTIENEntities db = new BANTIENEntities();

// GET: SanPhams

public ActionResult Index(string searchString)

{

var listKH = db.SanPhams.ToList();

if (!String.IsNullOrEmpty(searchString))

{

var links = db.SanPhams.Where(s => s.MaTien.ToString().Contains(searchString)

|| s.TenTien.Contains(searchString));

return View(links.ToList());

}

else

return View(listKH);

}

// GET: SanPhams/Details/5

public ActionResult Details(int? id)

{

if (id == null)

{

return new HttpStatusCodeResult(HttpStatusCode.BadRequest);

}

SanPham sanPham = db.SanPhams.Find(id);

if (sanPham == null)

{

return HttpNotFound();

}

return View(sanPham);

}

// GET: SanPhams/Create

public ActionResult Create()

{

ViewBag.TheLoai = new SelectList(db.LoaiTiens, "MaLoai", "TenLoai");

return View();

}

// POST: SanPhams/Create

// To protect from overposting attacks, enable the specific properties you want to bind to, for

// more details see https://go.microsoft.com/fwlink/?LinkId=317598.

[HttpPost]

[ValidateAntiForgeryToken]

public ActionResult Create([Bind(Include = "MaTien,TenTien,TacGia,NhaXuatBan,NgayXuatBan,TheLoai,SoLuong,DonGia,AnhMinhHoa")] SanPham sanPham)

{

if (ModelState.IsValid)

{

db.SanPhams.Add(sanPham);

db.SaveChanges();

return RedirectToAction("Index");

}

ViewBag.TheLoai = new SelectList(db.LoaiTiens, "MaLoai", "TenLoai", sanPham.TheLoai);

return View(sanPham);

}

// GET: SanPhams/Edit/5

public ActionResult Edit(int? id)

{

if (id == null)

{

return new HttpStatusCodeResult(HttpStatusCode.BadRequest);

}

SanPham sanPham = db.SanPhams.Find(id);

if (sanPham == null)

{

return HttpNotFound();

}

ViewBag.TheLoai = new SelectList(db.LoaiTiens, "MaLoai", "TenLoai", sanPham.TheLoai);

return View(sanPham);

}

// POST: SanPhams/Edit/5

// To protect from overposting attacks, enable the specific properties you want to bind to, for

// more details see https://go.microsoft.com/fwlink/?LinkId=317598.

[HttpPost]

[ValidateAntiForgeryToken]

public ActionResult Edit([Bind(Include = "MaTien,TenTien,TacGia,NhaXuatBan,NgayXuatBan,TheLoai,SoLuong,DonGia,AnhMinhHoa")] SanPham sanPham)

{

if (ModelState.IsValid)

{

db.Entry(sanPham).State = EntityState.Modified;

db.SaveChanges();

return RedirectToAction("Index");

}

ViewBag.TheLoai = new SelectList(db.LoaiTiens, "MaLoai", "TenLoai", sanPham.TheLoai);

return View(sanPham);

}

// GET: SanPhams/Delete/5

public ActionResult Delete(int? id)

{

if (id == null)

{

return new HttpStatusCodeResult(HttpStatusCode.BadRequest);

}

SanPham sanPham = db.SanPhams.Find(id);

if (sanPham == null)

{

return HttpNotFound();

}

return View(sanPham);

}

// POST: SanPhams/Delete/5

[HttpPost, ActionName("Delete")]

[ValidateAntiForgeryToken]

public ActionResult DeleteConfirmed(int id)

{

SanPham sanPham = db.SanPhams.Find(id);

db.SanPhams.Remove(sanPham);

db.SaveChanges();

return RedirectToAction("Index");

}

protected override void Dispose(bool disposing)

{

if (disposing)

{

db.Dispose();

}

base.Dispose(disposing);

}

}

}

* 1. Code user

#### AccountController

using System;

using System.Collections.Generic;

using System.Data.Entity;

using System.Linq;

using System.Web.Mvc;

using System.Web.Security;

using BanTien.Models;

namespace BanTien.Controllers

{

public class AccountController : Controller

{

BANTIENEntities db = new BANTIENEntities();

// GET: Account

[Authorize]

[HttpGet]

public ActionResult \_Profile(string username)

{

ProfileModel pro = new ProfileModel();

pro.username = username;

pro.Email = (from u in db.TaiKhoans

where u.Username == username

select u.Email).FirstOrDefault();

var check = db.KhachHangs.FirstOrDefault(x => x.UsernameKH == username);

if(check != null)

{

pro.HoTen = check.HoKH + " " + check.TenKH;

pro.SDT = check.DienThoai;

pro.DiaChi = check.DiaChi;

pro.GioiTinh = check.GioiTinh;

pro.avatar = check.AnhDaiDien;

pro.SinhNhat = check.NgaySinh;

}

else

{

var ad = db.Admins.FirstOrDefault(x => x.UsernameAD == username);

pro.HoTen = ad.HoAD + " " + ad.TenAD;

pro.SDT = ad.DienThoai;

pro.DiaChi = ad.DiaChi;

pro.GioiTinh = ad.GioiTinh;

pro.avatar = ad.AnhDaiDien;

pro.SinhNhat = ad.NgaySinh;

}

return View(pro);

}

[HttpGet]

[AllowAnonymous]

public ActionResult Login()

{

return View();

}

[HttpPost]

[ValidateAntiForgeryToken]

[AllowAnonymous]

public ActionResult Login(LoginModel model)

{

TaiKhoan taiKhoan = db.TaiKhoans.FirstOrDefault(x => x.Username == model.UsernameOrEmail);

if (taiKhoan == null)

{

taiKhoan = db.TaiKhoans.FirstOrDefault(x => x.Email == model.UsernameOrEmail);

if (taiKhoan == null)

{

ModelState.AddModelError("Invalid", "Invalid username or email");

return View();

}

goto Success;

}

Success:

if (taiKhoan.Password != model.Password)

{

ModelState.AddModelError("Invalid", "Incorrect password");

return View();

}

if (taiKhoan.block == true)

{

return RedirectToAction("Error", "Home");

}

FormsAuthentication.SetAuthCookie(taiKhoan.Username, false);

Session["user"] = taiKhoan.Username;

Session["role"] = taiKhoan.Role;

taiKhoan.Latest\_Login = DateTime.Now;

db.Entry(taiKhoan).State = EntityState.Modified;

db.SaveChanges();

return RedirectToAction("Index", "Home");

}

[Authorize]

public ActionResult Logout()

{

FormsAuthentication.SignOut();

Session.Abandon();

return RedirectToAction("Index", "Home");

}

/\* Register \*/

https://i.imgur.com/G5scMi1.pnghttps://i.imgur.com/G5scMi1.pnghttps://i.imgur.com/G5scMi1.png[HttpGet]

[AllowAnonymous]

public ActionResult Register()

{

return View();

}

// Post

[HttpPost]

[ValidateAntiForgeryToken]

[AllowAnonymous]

public ActionResult Register(RegisterModel model)

{

TaiKhoan taiKhoan = new TaiKhoan();

bool exists = db.TaiKhoans.Any(x => x.Username == model.Username);

if (exists)

{

ModelState.AddModelError("Invalid", "Username Exists");

return View();

}

exists = db.TaiKhoans.Any(x => x.Email == model.Email);

if (exists)

{

ModelState.AddModelError("Invalid", "Email Exists");

return View();

}

if (!(model.Password.Equals(model.ConfirmPassword)))

{

ModelState.AddModelError("Invalid", "Password not match");

return View();

}

taiKhoan.Username = model.Username;

taiKhoan.Password = model.Password;

taiKhoan.Email = model.Email;

taiKhoan.block = false;

taiKhoan.Role = "user";

taiKhoan.Latest\_Login = DateTime.Now;

db.TaiKhoans.Add(taiKhoan);

db.SaveChanges();

Session["username"] = model.Username;

Session["role"] = "user";

FormsAuthentication.SetAuthCookie(model.Username, false);

return RedirectToAction("InfoRegister", "KhachHang", new { username = model.Username });

}

/\* Forgot password \*/

[HttpGet]

public ActionResult ForgotPassword()

{

return View();

}

// Post

[HttpPost]

public ActionResult ForgotPassword(string UsernameOrEmail)

{

TaiKhoan myUser = db.TaiKhoans.FirstOrDefault(u => u.Username == UsernameOrEmail);

if (myUser == null)

{

myUser = db.TaiKhoans.FirstOrDefault(u => u.Email == UsernameOrEmail);

if (myUser == null)

{

ModelState.AddModelError("Invalid", "Invalid username or email");

return View();

}

goto Success;

}

Success:

return RedirectToAction("ResetPassword", "Account", new { username = myUser.Username });

}

[HttpGet]

public ActionResult ResetPassword(string username)

{

ViewBag.username = username;

return View();

}

[HttpPost]

[ValidateAntiForgeryToken]

public ActionResult ResetPassword(string Username, string Password, string ConfirmPassword)

{

var taiKhoan = db.TaiKhoans.Find(Username);

if (!(Password.Equals(ConfirmPassword)))

{

ModelState.AddModelError("Invalid", "Password not match");

return View();

}

taiKhoan.Password = Password;

db.Entry(taiKhoan).State = System.Data.Entity.EntityState.Modified;

db.SaveChanges();

return RedirectToAction("Login", "Account");

}

[HttpGet]

[Authorize]

public ActionResult ChangePassword(string username)

{

ViewBag.username = username;

return View();

}

[Authorize]

[HttpPost]

public ActionResult ChangePassword(string username, string password, string new\_pass, string confirm)

{

var user = db.TaiKhoans.Find(username);

if(user.Password != password)

{

ModelState.AddModelError("Invalid", "Incorrect password");

return View();

}

if (new\_pass != confirm)

{

ModelState.AddModelError("Invalid", "Password not match");

return View();

}

user.Password = new\_pass;

db.Entry(user).State = System.Data.Entity.EntityState.Modified;

db.SaveChanges();

return RedirectToAction("Index", "Home");

}

[Authorize]

public ActionResult Block(string username)

{

var taiKhoan = db.TaiKhoans.Find(username);

if (taiKhoan.block == true)

taiKhoan.block = false;

else

taiKhoan.block = true;

db.Entry(taiKhoan).State = System.Data.Entity.EntityState.Modified;

db.SaveChanges();

return RedirectToAction("Index", "KhachHangs");

}

[HttpGet]

public ActionResult Edit(string username)

{

EditAccountModel pro = new EditAccountModel();

pro.username = username;

pro.Email = (from u in db.TaiKhoans

where u.Username == username

select u.Email).FirstOrDefault();

var check = db.KhachHangs.FirstOrDefault(x => x.UsernameKH == username);

if (check != null)

{

pro.Ho\_tendem = check.HoKH;

pro.Ten = check.TenKH;

pro.SDT = check.DienThoai;

pro.DiaChi = check.DiaChi;

pro.GioiTinh = check.GioiTinh;

pro.SinhNhat = check.NgaySinh;

}

else

{

var ad = db.Admins.FirstOrDefault(x => x.UsernameAD == username);

pro.Ho\_tendem = ad.HoAD;

pro.Ten = ad.TenAD;

pro.SDT = ad.DienThoai;

pro.DiaChi = ad.DiaChi;

pro.GioiTinh = ad.GioiTinh;

pro.SinhNhat = ad.NgaySinh;

}

return View(pro);

}

[HttpPost]

public ActionResult Edit(EditAccountModel model)

{

var tk = db.TaiKhoans.Find(model.username);

tk.Email = model.Email;

db.Entry(tk).State = System.Data.Entity.EntityState.Modified;

var check = db.KhachHangs.FirstOrDefault(x => x.UsernameKH == model.username);

if (check != null)

{

check.HoKH = model.Ho\_tendem;

check.TenKH = model.Ten;

check.DienThoai = model.SDT;

check.DiaChi = model.DiaChi;

check.GioiTinh = model.GioiTinh;

check.NgaySinh = model.SinhNhat;

db.Entry(check).State = System.Data.Entity.EntityState.Modified;

}

else

{

var ad = db.Admins.FirstOrDefault(x => x.UsernameAD == model.username);

ad.HoAD = model.Ho\_tendem;

ad.TenAD = model.Ten;

ad.DienThoai = model.SDT;

ad.DiaChi = model.DiaChi;

ad.GioiTinh = model.GioiTinh;

ad.NgaySinh = model.SinhNhat;

db.Entry(ad).State = System.Data.Entity.EntityState.Modified;

}

db.SaveChanges();

return RedirectToAction("\_Profile", "Account", new { username = model.username });

}

}

}

#### DanhGiaController

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.Mvc;

using BanTien.Models;

namespace BanTien.Controllers

{

public class DanhGiaController : Controller

{

BANTIENEntities db = new BANTIENEntities();

[HttpPost]

public ActionResult Add(int id\_khach, int id\_sach, string review)

{

DanhGiaTien dg = new DanhGiaTien();

dg.MaKH = id\_khach;

dg.MaTien = id\_sach;

dg.NoiDungDanhGia = review;

dg.ThoiGianDang = DateTime.Now;

db.DanhGiaTiens.Add(dg);

db.SaveChanges();

return RedirectToAction("Details", "SanPhams", new { id = id\_sach });

}

[HttpPost]

public ActionResult Delete(int id\_khach, int id\_sach)

{

var dg = (from u in db.DanhGiaTiens

where u.MaKH == id\_khach && u.MaTien == id\_sach

select u).FirstOrDefault();

db.DanhGiaTiens.Remove(dg);

db.SaveChanges();

return RedirectToAction("Details", "SanPhams", new { id = id\_sach });

}

}

}

#### GioHangController

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.Mvc;

using BanTien.Models;

using System.Net;

using System.Net.Mail;

using System.Data.Entity;

namespace BanTien.Controllers

{

public class GioHangController : Controller

{

BANTIENEntities db = new BANTIENEntities();

/\* Get all cart items of this customer \*/

[HttpGet]

[Authorize]

public ActionResult Index(int? id)

{

if (id == null)

{

return new HttpStatusCodeResult(HttpStatusCode.BadRequest);

}

ViewBag.kh\_id = id;

var listItems = (from u in db.GioHangs

where u.MaKH == id && u.MaDH == null

select u).ToList();

if (listItems == null)

{

return new HttpStatusCodeResult(HttpStatusCode.BadRequest);

}

return View(listItems);

}

[HttpGet]

[Authorize]

public ActionResult ThanhToan(int kh)

{

var khachhang = db.KhachHangs.Find(kh);

var giohang = (from u in db.GioHangs

where u.MaKH == kh && u.MaDH == null

select u).ToList();

ProceedPaymentModel model = new ProceedPaymentModel();

model.MaKH = kh;

model.Ho\_TenKH = khachhang.HoKH + " " + khachhang.TenKH;

model.SDTDatHang = khachhang.DienThoai;

model.DiaChiNhanHang = khachhang.DiaChi;

model.EmailXacNhan = khachhang.TaiKhoan.Email;

model.TongTien = 0;

model.SachDaMua = new List<ItemPayment>();

for (int i = 0; i < giohang.Count; i++)

{

ItemPayment item = new ItemPayment();

item.STT = i + 1;

item.TenTien = giohang[i].SanPham.TenTien;

item.SoLuong = giohang[i].SoLuong;

item.DonGia = giohang[i].SanPham.DonGia;

item.ThanhTien = giohang[i].ThanhTien;

model.SachDaMua.Add(item);

model.TongTien += giohang[i].ThanhTien;

}

model.GiamGia = khachhang.GiamGia;

model.TongTien -= model.TongTien \* ((double)model.GiamGia / 100);

model.TongThanhToan = model.TongTien + 15000;

return View(model);

}

[ChildActionOnly]

public ActionResult RenderInvoice(List<GioHang> model, int id)

{

PreReceipt pre = new PreReceipt();

if (model.Count == 0)

{

pre.Total\_Item = 0;

pre.Discount = getDiscount(id);

pre.Total\_Quantity = 0;

pre.Provisional = 0;

pre.Total\_Money = 0;

goto Success;

}

pre.Total\_Item = model.Count(x => x.MaKH == id);

pre.Discount = getDiscount(id);

pre.Total\_Quantity = 0;

pre.Provisional = 0;

for (int i = 0; i < pre.Total\_Item; i++)

{

pre.Total\_Quantity += model[i].SoLuong;

pre.Provisional += model[i].ThanhTien;

}

pre.Total\_Money = pre.Provisional - pre.Provisional \* ((double)pre.Discount / 100);

Success:

ViewBag.khachhang\_thanhtoan\_id = id;

return PartialView("~/Views/Shared/\_InvoicePreview.cshtml", pre);

}

public int? getDiscount(int id)

{

return (from u in db.KhachHangs

where u.MaKH == id

select u.GiamGia).FirstOrDefault();

}

/\* Add an item to MaKH's cart \*/

[Authorize]

[HttpPost]

public ActionResult AddItem(string MaKH, string MaTien, string quantity, string controller, string action)

{

int kh = Int32.Parse(MaKH);

int sach = Int32.Parse(MaTien);

int soluong = Int32.Parse(quantity);

// Check quantity in storage

var sp = db.SanPhams.Find(sach);// Find the product

if(sp.SoLuong < soluong)

{

return RedirectToAction(action, controller);

}

// Check exist in MaKH's cart

if(existInCart(kh,sach) != null)

{

existInCart(kh, sach).Thoigian = DateTime.Now;

existInCart(kh, sach).SoLuong += soluong;

existInCart(kh, sach).ThanhTien += soluong \* sp.DonGia;

db.Entry(existInCart(kh, sach)).State = EntityState.Modified;

}

else

{

GioHang gh = new GioHang();

gh.MaKH = kh;

gh.MaTien = sach;

gh.SoLuong = soluong;

gh.Thoigian = DateTime.Now;

gh.ThanhTien = soluong \* sp.DonGia;

db.GioHangs.Add(gh);

}

sp.SoLuong -= soluong;

db.Entry(sp).State = EntityState.Modified;

db.SaveChanges();

return RedirectToAction(action, controller);

}

/\* check if item exists in MaKH's cart \*/

[NonAction]

public GioHang existInCart(int kh, int sach)

{

return (from u in db.GioHangs

where u.MaKH == kh && u.MaTien == sach && u.MaDH == null

select u).FirstOrDefault();

}

[Authorize]

[HttpPost]

public ActionResult ModifyQuantity(int gh, int kh, int quantity, double dongia)

{

var cart = db.GioHangs.Find(gh);

var sp = db.SanPhams.Find(cart.MaTien);

sp.SoLuong -= (quantity - cart.SoLuong);

cart.SoLuong += (quantity - cart.SoLuong);

cart.ThanhTien = cart.SoLuong \* dongia;

cart.Thoigian = DateTime.Now;

db.Entry(cart).State = EntityState.Modified;

db.Entry(sp).State = EntityState.Modified;

db.SaveChanges();

return RedirectToAction("Index", "GioHang", new { id = kh });

}

[Authorize]

[HttpPost]

public ActionResult Delete(int sa, int gh, int kh)

{

var cart = db.GioHangs.Find(gh);

var sp = db.SanPhams.Find(sa);

sp.SoLuong += cart.SoLuong;

db.Entry(sp).State = EntityState.Modified;

db.GioHangs.Remove(cart);

db.SaveChanges();

return RedirectToAction("Index", "GioHang", new { id = kh });

}

}

}

#### HomeController

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.Mvc;

using System.Data.Entity;

using BanTien.Models;

using System.Collections;

using PagedList;

namespace BanTien.Controllers

{

public class HomeController : Controller

{

BANTIENEntities db = new BANTIENEntities();

[HttpGet]

public ActionResult Index(int? page, int? type, string kind, string search\_text, string filter, string price\_range) // type to choose a search kind to bind data

{

if (page == null)

page = 1;

if (type == null) // set type = 1 when page starts

type = 0;

int pageSize = 15;

List<SanPham> sanPham = new List<SanPham>();

//

switch(type)

{

case 0: // page starts

sanPham = (from a in db.SanPhams

select a).ToList(); break;

case 1:

{

if (kind == "ALL")

sanPham = (from a in db.SanPhams

select a).ToList();

else

sanPham = Kind\_Search(kind);

} break;

case 2:

{

if (search\_text == null)

sanPham = (from a in db.SanPhams

select a).ToList();

else

sanPham = Text\_Search(search\_text, filter);

}

break;

case 3: sanPham = Price\_Search(price\_range); break;

default:break;

}

ViewBag.type = type;

ViewBag.price\_range = price\_range;

ViewBag.kind = kind;

ViewBag.search\_text = search\_text;

ViewBag.filter = filter;

ViewBag.itemCount = sanPham.Count;

int pageNumber = (page ?? 1);

return View(sanPham.ToPagedList(pageNumber, pageSize));

}

[HttpPost]

public ActionResult Index(string kind, string search\_text, string filter, string price\_range) // type to choose a search kind to bind data

{

if (kind != null)

return RedirectToAction("Index", new { type = 1, kind = kind});

if (price\_range != null)

return RedirectToAction("Index", new { type = 3, price\_range = price\_range });

if (search\_text != null)

return RedirectToAction("Index", new { type = 2, search\_text = search\_text, filter = filter });

return View();

}

[NonAction]

public List<SanPham> Kind\_Search(string kind)

{

return (from u in db.SanPhams

where u.TheLoai == kind

select u).ToList();

}

[NonAction]

public List<SanPham> Price\_Search(string price\_range)

{

switch (price\_range)

{

case "<1000":

return (from a in db.SanPhams

where a.DonGia < 1000000

select a).ToList();

case "1000-3000":

return (from a in db.SanPhams

where a.DonGia > 1000000 && a.DonGia < 3000000

select a).ToList();

case ">3000":

return (from a in db.SanPhams

where a.DonGia > 3000000

select a).ToList();

default: return null;

}

}

[NonAction]

public List<SanPham> Text\_Search(string search\_text, string filter)

{

if (search\_text == "")

return null;

else

{

search\_text = search\_text.ToLower();

switch(filter)

{

case "name":

return (from u in db.SanPhams

where u.TenTien.ToLower().Contains(search\_text)

select u).ToList();

case "category":

return (from u in db.SanPhams

where u.LoaiTien.TenLoai.ToLower().Contains(search\_text)

select u).ToList();

default: return null;

}

}

}

public ActionResult Error()

{

return View();

}

public ActionResult About()

{

return View();

}

public ActionResult Feedback()

{

return View();

}

public ActionResult Support()

{

return View();

}

public ActionResult Policy()

{

ViewBag.Message = "Your policy page.";

return View();

}

public ActionResult Revenue()

{

return View();

}

}

}

#### KhachHangController

using BanTien.Models;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Net;

using System.Web;

using System.Web.Mvc;

namespace BanTien.Controllers

{

public class KhachHangController : Controller

{

BANTIENEntities db = new BANTIENEntities();

/\* Update info right after registration \*/

[HttpGet]

[Authorize]

public ActionResult InfoRegister(string username)

{

if (username.Equals(""))

return new HttpStatusCodeResult(HttpStatusCode.BadRequest);

ViewBag.username = username;

return View();

}

[HttpPost]

[ValidateAntiForgeryToken]

[Authorize]

public ActionResult InfoRegister(CustomerModel customer)

{

KhachHang khachHang = new KhachHang();

khachHang.HoKH = customer.LastName;

khachHang.TenKH = customer.FirstName;

khachHang.DienThoai = customer.Phone;

khachHang.GioiTinh = customer.Gender;

khachHang.DiaChi = customer.Address;

khachHang.NgaySinh = customer.Birthday;

khachHang.AnhDaiDien = "/images/avatar\_icon.png";

khachHang.SoLanHuyDon = 0;

khachHang.GiamGia = 5;

khachHang.UsernameKH = Session["username"].ToString();

db.KhachHangs.Add(khachHang);

db.SaveChanges();

return RedirectToAction("Index", "Home");

}

}

}

1. Phân tích CSDL
   1. Query

USE [master]

GO

/\*\*\*\*\*\* Object: Database [BANTIEN] Script Date: 12/13/2022 6:26:54 AM \*\*\*\*\*\*/

CREATE DATABASE [BANTIEN]

CONTAINMENT = NONE

ON PRIMARY

( NAME = N'BANTIEN', FILENAME = N'C:\Program Files\Microsoft SQL Server\MSSQL15.MSSQLSERVER\MSSQL\DATA\BANTIEN.mdf' , SIZE = 8192KB , MAXSIZE = UNLIMITED, FILEGROWTH = 65536KB )

LOG ON

( NAME = N'BANTIEN\_log', FILENAME = N'C:\Program Files\Microsoft SQL Server\MSSQL15.MSSQLSERVER\MSSQL\DATA\BANTIEN\_log.ldf' , SIZE = 8192KB , MAXSIZE = 2048GB , FILEGROWTH = 65536KB )

WITH CATALOG\_COLLATION = DATABASE\_DEFAULT

GO

ALTER DATABASE [BANTIEN] SET COMPATIBILITY\_LEVEL = 150

GO

IF (1 = FULLTEXTSERVICEPROPERTY('IsFullTextInstalled'))

begin

EXEC [BANTIEN].[dbo].[sp\_fulltext\_database] @action = 'enable'

end

GO

ALTER DATABASE [BANTIEN] SET ANSI\_NULL\_DEFAULT OFF

GO

ALTER DATABASE [BANTIEN] SET ANSI\_NULLS OFF

GO

ALTER DATABASE [BANTIEN] SET ANSI\_PADDING OFF

GO

ALTER DATABASE [BANTIEN] SET ANSI\_WARNINGS OFF

GO

ALTER DATABASE [BANTIEN] SET ARITHABORT OFF

GO

ALTER DATABASE [BANTIEN] SET AUTO\_CLOSE ON

GO

ALTER DATABASE [BANTIEN] SET AUTO\_SHRINK OFF

GO

ALTER DATABASE [BANTIEN] SET AUTO\_UPDATE\_STATISTICS ON

GO

ALTER DATABASE [BANTIEN] SET CURSOR\_CLOSE\_ON\_COMMIT OFF

GO

ALTER DATABASE [BANTIEN] SET CURSOR\_DEFAULT GLOBAL

GO

ALTER DATABASE [BANTIEN] SET CONCAT\_NULL\_YIELDS\_NULL OFF

GO

ALTER DATABASE [BANTIEN] SET NUMERIC\_ROUNDABORT OFF

GO

ALTER DATABASE [BANTIEN] SET QUOTED\_IDENTIFIER OFF

GO

ALTER DATABASE [BANTIEN] SET RECURSIVE\_TRIGGERS OFF

GO

ALTER DATABASE [BANTIEN] SET ENABLE\_BROKER

GO

ALTER DATABASE [BANTIEN] SET AUTO\_UPDATE\_STATISTICS\_ASYNC OFF

GO

ALTER DATABASE [BANTIEN] SET DATE\_CORRELATION\_OPTIMIZATION OFF

GO

ALTER DATABASE [BANTIEN] SET TRUSTWORTHY OFF

GO

ALTER DATABASE [BANTIEN] SET ALLOW\_SNAPSHOT\_ISOLATION OFF

GO

ALTER DATABASE [BANTIEN] SET PARAMETERIZATION SIMPLE

GO

ALTER DATABASE [BANTIEN] SET READ\_COMMITTED\_SNAPSHOT OFF

GO

ALTER DATABASE [BANTIEN] SET HONOR\_BROKER\_PRIORITY OFF

GO

ALTER DATABASE [BANTIEN] SET RECOVERY SIMPLE

GO

ALTER DATABASE [BANTIEN] SET MULTI\_USER

GO

ALTER DATABASE [BANTIEN] SET PAGE\_VERIFY CHECKSUM

GO

ALTER DATABASE [BANTIEN] SET DB\_CHAINING OFF

GO

ALTER DATABASE [BANTIEN] SET FILESTREAM( NON\_TRANSACTED\_ACCESS = OFF )

GO

ALTER DATABASE [BANTIEN] SET TARGET\_RECOVERY\_TIME = 60 SECONDS

GO

ALTER DATABASE [BANTIEN] SET DELAYED\_DURABILITY = DISABLED

GO

ALTER DATABASE [BANTIEN] SET ACCELERATED\_DATABASE\_RECOVERY = OFF

GO

ALTER DATABASE [BANTIEN] SET QUERY\_STORE = OFF

GO

USE [BANTIEN]

GO

/\*\*\*\*\*\* Object: Table [dbo].[Admin] Script Date: 12/13/2022 6:26:54 AM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[Admin](

[MaAD] [int] IDENTITY(1,1) NOT NULL,

[HoAD] [nvarchar](10) NULL,

[TenAD] [nvarchar](30) NULL,

[UsernameAD] [varchar](50) NULL,

[GioiTinh] [char](1) NULL,

[DiaChi] [nvarchar](100) NULL,

[AnhDaiDien] [varchar](50) NULL,

[DienThoai] [varchar](10) NULL,

[NgaySinh] [date] NULL,

CONSTRAINT [PK\_Table\_1] PRIMARY KEY CLUSTERED

(

[MaAD] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [dbo].[DanhGiaTien] Script Date: 12/13/2022 6:26:55 AM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[DanhGiaTien](

[MaTien] [int] NOT NULL,

[MaKH] [int] NOT NULL,

[NoiDungDanhGia] [nvarchar](200) NULL,

[ThoiGianDang] [datetime] NULL,

CONSTRAINT [PK\_DanhGiaTien] PRIMARY KEY CLUSTERED

(

[MaTien] ASC,

[MaKH] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [dbo].[GioHang] Script Date: 12/13/2022 6:26:55 AM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[GioHang](

[MaGH] [int] IDENTITY(1,1) NOT NULL,

[MaKH] [int] NULL,

[MaTien] [int] NULL,

[SoLuong] [int] NULL,

[Thoigian] [datetime] NULL,

[ThanhTien] [float] NULL,

[MaDH] [int] NULL,

CONSTRAINT [PK\_GioHang] PRIMARY KEY CLUSTERED

(

[MaGH] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [dbo].[HoaDon] Script Date: 12/13/2022 6:26:55 AM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[HoaDon](

[MaHD] [int] IDENTITY(1,1) NOT NULL,

[MaKH] [int] NULL,

[NgayLapHD] [datetime] NULL,

[TongTien] [float] NULL,

[TenKhach] [nvarchar](50) NULL,

[SDTDatHang] [char](10) NULL,

[EmailXacNhan] [varchar](50) NULL,

[DiaChiNhanHang] [nvarchar](200) NULL,

[GiamGia] [int] NULL,

[TongThanhToan] [float] NULL,

[TrangThai] [nvarchar](50) NULL,

[NgayNhanHang] [date] NULL,

CONSTRAINT [PK\_HoaDon] PRIMARY KEY CLUSTERED

(

[MaHD] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [dbo].[KhachHang] Script Date: 12/13/2022 6:26:55 AM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[KhachHang](

[MaKH] [int] IDENTITY(1,1) NOT NULL,

[HoKH] [nvarchar](20) NULL,

[TenKH] [nvarchar](30) NULL,

[UsernameKH] [varchar](50) NULL,

[GioiTinh] [char](1) NULL,

[DiaChi] [nvarchar](100) NULL,

[AnhDaiDien] [varchar](50) NULL,

[DienThoai] [varchar](10) NULL,

[NgaySinh] [date] NULL,

[GiamGia] [int] NULL,

[SoLanHuyDon] [smallint] NULL,

CONSTRAINT [PK\_KhachHang] PRIMARY KEY CLUSTERED

(

[MaKH] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [dbo].[LoaiTien] Script Date: 12/13/2022 6:26:55 AM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[LoaiTien](

[MaLoai] [varchar](50) NOT NULL,

[TenLoai] [nvarchar](50) NOT NULL,

CONSTRAINT [PK\_LoaiTien\_1] PRIMARY KEY CLUSTERED

(

[MaLoai] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [dbo].[SanPham] Script Date: 12/13/2022 6:26:55 AM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[SanPham](

[MaTien] [int] IDENTITY(1,1) NOT NULL,

[TenTien] [nvarchar](100) NULL,

[MoTaTien] [varchar](50) NULL,

[TheLoai] [varchar](50) NULL,

[SoLuong] [int] NULL,

[DonGia] [float] NULL,

[AnhMinhHoa] [varchar](60) NULL,

CONSTRAINT [PK\_SanPham] PRIMARY KEY CLUSTERED

(

[MaTien] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [dbo].[TaiKhoan] Script Date: 12/13/2022 6:26:55 AM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[TaiKhoan](

[Username] [varchar](50) NOT NULL,

[Password] [varchar](100) NULL,

[Role] [varchar](10) NULL,

[Email] [varchar](50) NULL,

[Latest\_Login] [datetime] NULL,

[block] [bit] NULL,

CONSTRAINT [PK\_TaiKhoan] PRIMARY KEY CLUSTERED

(

[Username] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

GO

ALTER TABLE [dbo].[Admin] WITH CHECK ADD CONSTRAINT [FK\_Admin\_TaiKhoan] FOREIGN KEY([UsernameAD])

REFERENCES [dbo].[TaiKhoan] ([Username])

ON UPDATE CASCADE

ON DELETE CASCADE

GO

ALTER TABLE [dbo].[Admin] CHECK CONSTRAINT [FK\_Admin\_TaiKhoan]

GO

ALTER TABLE [dbo].[DanhGiaTien] WITH CHECK ADD CONSTRAINT [FK\_DanhGiaTien\_KhachHang] FOREIGN KEY([MaKH])

REFERENCES [dbo].[KhachHang] ([MaKH])

ON UPDATE CASCADE

ON DELETE CASCADE

GO

ALTER TABLE [dbo].[DanhGiaTien] CHECK CONSTRAINT [FK\_DanhGiaTien\_KhachHang]

GO

ALTER TABLE [dbo].[DanhGiaTien] WITH CHECK ADD CONSTRAINT [FK\_DanhGiaTien\_SanPham] FOREIGN KEY([MaTien])

REFERENCES [dbo].[SanPham] ([MaTien])

ON UPDATE CASCADE

ON DELETE CASCADE

GO

ALTER TABLE [dbo].[DanhGiaTien] CHECK CONSTRAINT [FK\_DanhGiaTien\_SanPham]

GO

ALTER TABLE [dbo].[GioHang] WITH CHECK ADD CONSTRAINT [FK\_GioHang\_HoaDon1] FOREIGN KEY([MaDH])

REFERENCES [dbo].[HoaDon] ([MaHD])

ON UPDATE CASCADE

ON DELETE CASCADE

GO

ALTER TABLE [dbo].[GioHang] CHECK CONSTRAINT [FK\_GioHang\_HoaDon1]

GO

ALTER TABLE [dbo].[GioHang] WITH CHECK ADD CONSTRAINT [FK\_GioHang\_KhachHang] FOREIGN KEY([MaKH])

REFERENCES [dbo].[KhachHang] ([MaKH])

ON UPDATE CASCADE

ON DELETE CASCADE

GO

ALTER TABLE [dbo].[GioHang] CHECK CONSTRAINT [FK\_GioHang\_KhachHang]

GO

ALTER TABLE [dbo].[GioHang] WITH CHECK ADD CONSTRAINT [FK\_GioHang\_SanPham] FOREIGN KEY([MaTien])

REFERENCES [dbo].[SanPham] ([MaTien])

ON UPDATE CASCADE

ON DELETE CASCADE

GO

ALTER TABLE [dbo].[GioHang] CHECK CONSTRAINT [FK\_GioHang\_SanPham]

GO

ALTER TABLE [dbo].[KhachHang] WITH CHECK ADD CONSTRAINT [FK\_KhachHang\_TaiKhoan] FOREIGN KEY([UsernameKH])

REFERENCES [dbo].[TaiKhoan] ([Username])

ON UPDATE CASCADE

ON DELETE CASCADE

GO

ALTER TABLE [dbo].[KhachHang] CHECK CONSTRAINT [FK\_KhachHang\_TaiKhoan]

GO

ALTER TABLE [dbo].[SanPham] WITH CHECK ADD CONSTRAINT [FK\_SanPham\_LoaiTien] FOREIGN KEY([TheLoai])

REFERENCES [dbo].[LoaiTien] ([MaLoai])

ON UPDATE CASCADE

ON DELETE CASCADE

GO

ALTER TABLE [dbo].[SanPham] CHECK CONSTRAINT [FK\_SanPham\_LoaiTien]

GO

USE [master]

GO

ALTER DATABASE [BANTIEN] SET READ\_WRITE

GO

1. Kết luận

* Tính năng đã làm được
* Có sử dụng database vào phần login, sản phẩm, hóa đơn, vỏ hàng.
* Có thể sử dụng công cụ để tìm kiếm sản phẩm theo loại tiền hoặc tên tiền.
* Có tính năng theo giá tiền.
* Có tính năng khóa tài khoản nếu khách hàng hủy 4 đơn hàng liên tiếp
* Có thể xem được trạng thái đơn hàng.
* Có tính năng chỉnh sửa thông tin cá nhân.
* Có tính năng thống kê doanh thu
* Tính năng chưa làm được
* Chưa thể thanh toán bằng ví điện tử hoặc banking

1. Tài liệu tham khảo

1/ Slide bài giảng – Tưởng Thị Xuân Thu

2/ xuanthulab.net

3/ W3schol.com

4/ Gary Schneider-Electronic Commerce -Course Technology (2010)