# khung doi

**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TPHCM**

**KHOA ĐÀO TẠO CHẤT LƯỢNG CAO**



**ĐỒ ÁN CÔNG NGHỆ THÔNG TIN**

**THIẾT KẾ VÀ XÂY DỰNG**

**APP THƯƠNG MẠI ĐIỆN TỬ**

**Giáo viên hướng dẫn: Hoàng Văn Dũng**

**Nhóm sinh viên thực hiện: Nhóm 1**

|  |  |  |
| --- | --- | --- |
| **MSSV** | **Họ và tên** | **Mức độ đóng góp** |
| **20110103** | **Nguyễn Gia Huy** | **100%** |
| **20110493** | **Huỳnh Lê Huy** | **100%** |
| **20110069** | **Nguyễn Tường Phương Uyên** | **100%** |

**TPHCM, tháng 12 năm 2022**

# --------------------



**ĐỒ ÁN CÔNG NGHỆ THÔNG TIN**

**THIẾT KẾ VÀ XÂY DỰNG**

**APP THƯƠNG MẠI ĐIỆN TỬ**

**Giáo viên hướng dẫn: Hoàng Văn Dũng**

**Nhóm sinh viên thực hiện: Nhóm 1**

|  |  |  |
| --- | --- | --- |
| **MSSV** | **Họ và tên** | **Mức độ đóng góp** |
| **20110103** | **Nguyễn Gia Huy** | **100%** |
| **20110493** | **Huỳnh Lê Huy** | **100%** |
|  |  | **100%** |

**TPHCM, tháng 10 năm 2022**

# LỜI CẢM ƠN

Lời đầu tiên, chúng em xin gửi lời cảm ơn chân thành nhất đến giảng viên bộ môn PSG.TS thầy Hoàng Văn Dũng. Trong quá trình học tập và tìm hiểu cũng như làm đề tài, chúng đã nhận được sự quan tâm giúp đỡ, hướng dẫn rất tận tình, tâm huyết của thầy. Thầy đã giúp chúng em tích luỹ thêm nhiều kiến thức, định hướng cho chúng em trong suốt đề tài. Từ những kinh nghiệm tích luỹ được đã giúp chúng em có thể hoàn thành tốt đề tài đặt ra, khắc phục những điểm khiếm khuyết, từ đó hoàn thiện hơn trong những môn học tiếp theo.

Để có thể có đầy đủ lượng kiến thức cần thiết để hoàn thành đề tài này, nhóm cũng xin gửi lời cảm ơn chân thành đến các quý thầy cô trong khoa Đào tạo Chất Lượng Cao cũng như trong khoa Công Nghệ Thông Tin đã cung cấp những kiến thức cơ bản, nền tảng cho cả môn học này cũng như trong suốt những dự án sau này. Và để có thể giúp cho đề tài được đầy đủ hơn, nhóm cũng xin cảm ơn các bạn, các anh chị đã mang đến những lời góp ý để đề tài hoàn thiện hơn ạ.

Với chủ đề đề tài là một app thương mại điện tử cùng với chỉ thực hiện trong một khoảng thời gian tương đối ngắn ngủi, việc đề tài có những thiếu sót là điều khó tránh khỏi. Việc thiếu kinh nghiệm thực tiễn về mặt kiến thức lẫn kĩ thuật nên những ý kiến, phê bình quý báu của giảng viên sẽ giúp nhóm hoàn thiện hơn và cải thiện, khắc phục những hạn chế trong các dự án tiếp theo.

Lời nói cuối cùng, nhóm xin kính chúc quý thầy, cô dồi dào sức khoẻ và ngày càng phát triển trong sự nghiệp giáo dục, dưỡng dạy các thế hệ văn hoá và đầy kiến thức. Một lần nữa, chúng em xin chân thành cảm ơn ạ.

# MỤC LỤC

[MỤC LỤC 4](#_Toc120818182)

[MỤC LỤC BẢNG BIỂU 5](#_Toc120818183)

[MỤC LỤC HÌNH ẢNH 6](#_Toc120818184)

[CHƯƠNG 1: ĐẶC TẢ 8](#_Toc120818185)

[1.1. Phát biểu về đồ án 8](#_Toc120818186)

[1.2. Dữ liệu, thông tin đầu vào yêu cầu 8](#_Toc120818187)

[1.3. Mục đích, tính năng của ứng dụng 9](#_Toc120818188)

[1.4. Giao diện dự kiến 9](#_Toc120818189)

[1.4.1. Giao diện người dùng 10](#_Toc120818190)

[1.4.2. Giao diện người quản lý 10](#_Toc120818191)

[CHƯƠNG 2: PHÂN CÔNG CÔNG VIỆC 12](#_Toc120818192)

[CHƯƠNG 3: THIẾT KẾ ỨNG DỤNG 13](#_Toc120818193)

[3.1. Quá trình thiết kế 13](#_Toc120818194)

[3.2. Thiết kế lớp 13](#_Toc120818195)

[3.3. Thiết kế cơ sở dữ liệu 28](#_Toc120818196)

[3.3.1. Mối quan hệ giữa các bảng 28](#_Toc120818197)

[3.3.2. Mô tả các bảng 28](#_Toc120818198)

[3.4. Bảng mô tả các Fields 29](#_Toc120818199)

[3.5. Thiết kế giao diện 33](#_Toc120818200)

[CHƯƠNG 4: CÀI ĐẶT VÀ KIỂM THỬ 41](#_Toc120818201)

[CHƯƠNG 5: KẾT LUẬN 47](#_Toc120818202)

[5.1. Mức độ hoàn thành 47](#_Toc120818203)

[5.2. Các khó khăn và giải pháp 47](#_Toc120818204)

[5.3. Ưu điểm và hạn chế 47](#_Toc120818205)

[5.4. Ý tưởng phát triển đồ án 47](#_Toc120818206)

# MỤC LỤC BẢNG BIỂU

[Bảng 1: Phân công công việc 12](#_Toc120819470)

[Bảng 2A: Sinh viên phụ trách: Nguyễn Gia Huy (Các Class) 13](#_Toc120819471)

[Bảng 2B: Sinh viên phụ trách: Nguyễn Gia Huy (Các phương thức trong Class) 14](#_Toc120819472)

[Bảng 3A: Sinh viên phụ trách: Huỳnh Lê Huy (Các Class) 19](#_Toc120819473)

[Bảng 3B: Sinh viên phụ trách: Huỳnh Lê Huy (Các phương thức trong Class) 20](#_Toc120819474)

[Bảng 4A: Sinh viên phụ trách: Nguyễn Tường Phương Uyên (Các Class) 23](#_Toc120819475)

[Bảng 4B: Sinh viên phụ trách: Nguyễn Tường Phương Uyên (Các phương thức trong Class) 24](#_Toc120819476)

[Bảng 5: Bảng mô tả các bảng 28](#_Toc120819477)

[Bảng 6: Bảng mô tả table Users 29](#_Toc120819478)

[Bảng 7: Bảng mô tả table Product 29](#_Toc120819479)

[Bảng 8: Bảng mô tả table Product\_category 30](#_Toc120819480)

[Bảng 9: Bảng mô tả table Product\_Inventory 30](#_Toc120819481)

[Bảng 10: Bảng mô tả table User\_address 30](#_Toc120819482)

[Bảng 11: Bảng mô tả table User\_payment 31](#_Toc120819483)

[Bảng 12: Bảng mô tả table Payment\_details 31](#_Toc120819484)

[Bảng 13: Bảng mô tả table Order\_items 31](#_Toc120819485)

[Bảng 14: Bảng mô tả table Order\_details 32](#_Toc120819486)

[Bảng 15: Bảng mô tả table shopping\_session 32](#_Toc120819487)

[Bảng 16: Bảng mô tả table discount 32](#_Toc120819488)

[Bảng 17: Bảng mô tả table cart\_item 33](#_Toc120819489)

[Bảng 18: Bảng tổng hợp thiết kế giao diện người dùng 33](#_Toc120819490)

[Bảng 19: Bảng tổng hợp giao diện người quản lý 38](#_Toc120819491)

[Bảng 20: Bảng tình huống kiểm thử phần mềm 41](#_Toc120819492)

# MỤC LỤC HÌNH ẢNH

[Hình 1: Sơ đồ cơ sở dữ liệu 28](#_Toc120818135)

[Hình 2.1: Màn hình đăng nhập 33](#_Toc120818136)

[Hình 2.2: Màn hình đăng ký 34](#_Toc120818137)

[Hình 2.3: Màn hình Dashboard người dùng 34](#_Toc120818138)

[Hình 2.4: Màn hình Shopping Cart người dùng 35](#_Toc120818139)

[Hình 2.5: Giao diện chọn phương thức thanh toán 35](#_Toc120818140)

[Hình 2.6: Giao diện hiển thị đơn đặt hàng chưa thanh toán 35](#_Toc120818141)

[Hình 2.7: Giao diện chi tiết sản phẩm trong đơn hàng 35](#_Toc120818142)

[Hình 2.8: Giao diện hoá đơn người dùng đã thanh toán 36](#_Toc120818143)

[Hình 2.9: Giao diện thông tin cơ bản người dùng 36](#_Toc120818144)

[Hình 2.10: Giao diện thay đổi mật khẩu 36](#_Toc120818145)

[Hình 2.11: Giao diện thông tin thanh toán 37](#_Toc120818146)

[Hình 2.12: Giao diện thêm thông tin thanh toán 37](#_Toc120818147)

[Hình 3.1: Giao diện Dashboard người quản lý 38](#_Toc120818148)

[Hình 3.2: Giao diện All Products người quản lý 38](#_Toc120818149)

[Hình 3.3: Giao diện thêm sản phẩm người quản lý 38](#_Toc120818150)

[Hình 3.4: Giao diện Discount người quản lý 39](#_Toc120818151)

[Hình 3.5: Giao diện thêm Discount 39](#_Toc120818152)

[Hình 3.6: Giao diện Invoices người quản lý 39](#_Toc120818153)

[Hình 3.7: Giao diện User Accounts người quản lý 40](#_Toc120818154)

[Hình 4.1: Đăng ký người dùng mới 41](#_Toc120818155)

[Hình 4.2: Xem thử thông tin người dùng 41](#_Toc120818156)

[Hình 4.3: Thêm một số thông tin cơ bản 42](#_Toc120818157)

[Hình 4.4: Thêm thông tin thanh toán 42](#_Toc120818158)

[Hình 4.5: Tiến hành thêm vài sản phẩm vào giỏ hàng 42](#_Toc120818159)

[Hình 4.6: Tiến hành thanh toán bằng thẻ đã thêm 43](#_Toc120818160)

[Hình 4.7: Đơn hàng đã được đặt thành công 43](#_Toc120818161)

[Hình 4.8: Tiến hành vào Form Admin Complete đơn hàng 43](#_Toc120818162)

[Hình 4.9: Hoá đơn đã được thanh toán thành công 43](#_Toc120818163)

[Hình 4.10: Thông tin thống kê của người quản lý 44](#_Toc120818164)

[Hình 4.10: Thông tin thống kê về hoá đơn 44](#_Toc120818165)

[Hình 4.11: Thêm một discount mới vào hệ thống 45](#_Toc120818166)

[Hình 4.12: Thêm 3 sản phẩm vào discount và bật lên 45](#_Toc120818167)

[Hình 4.13: Giá sản phẩm được thêm đã thay đổi 45](#_Toc120818168)

[Hình 4.14: Tính toán phần discount trong giỏ hàng 45](#_Toc120818169)

[Hình 4.15: Danh sách các người dùng trong hệ thống 46](#_Toc120818170)

[Hình 4.16: Thay đổi mật khẩu cho người dùng 46](#_Toc120818171)

# CHƯƠNG 1: ĐẶC TẢ

## Phát biểu về đồ án

Hiện nay, trong một thời kì mà công nghệ đang bao phủ đến mọi ngóc ngách trong cuộc sống hằng ngày của chúng ta, từ việc thức dậy mỗi sáng cho đến những công việc bình thường nhất. Sự xuất hiện của công nghệ đã và đang mang lại một cách hoạt động rất khác của thế giới bởi sự tiện dụng và nhanh chóng mà nó mang lại. Hơn thế nữa, trong vài năm gần đây, việc đại dịch Covid-19 bùng nổ trên toàn thế giới khiến cho việc ra ngoài khó khăn hơn đã dẫn đến sự phát triển, bùng nổ của các dịch vụ thương mại điện tử trên toàn cầu. Đã và đang có rất nhiều hệ thống thương mại điện tử xuất hiện và phát triển nhanh chóng như Shopee, Lazada hay Tiki. Các hệ thống này hầu như hoạt động trên trình duyệt (browser) hay trên các application trên smartphone của người dùng. Và sử dụng chúng như một ứng dụng thông thường trên Windows là chưa thực sự phổ biến, việc sử dụng ứng dụng có thể mang lại sự tối ưu tốt hơn, phản hồi nhanh hơn và cả bảo mật tốt hơn. Đối với các nền tảng thương mại điện tử hiện nay, bảo mật dữ liệu nhạy cảm cho người dùng là một trong những vấn đề đáng quan tâm nhất và nếu ứng dụng được xây dựng dựa trên nền tảng hệ điều hành sẽ giúp tăng đáng kể khả năng bảo mật. Thế nên, nhóm chúng em đã quyết định chọn chủ đề cho đề tài môn học Đồ Án CNTT là “Xây dựng và thiết kế ứng dụng thương mại điện tử”. Mục tiêu của nhóm khi chọn chủ đề này là xây dựng một ứng dụng trên Windows để có thể giúp người dùng quản lý hệ thống thương mại điện tử để kinh doanh hoặc hướng tới những khách hàng muốn sử dụng ứng dụng để mua sắm nhanh chóng, tối ưu, bảo mật được dữ liệu quan trọng của bản thân. Phần mềm cũng hướng đến việc giúp cho người quản lý có thể quản lý được tình trạng kinh doanh cũng như thao tác đối với các tính năng cơ bản mà ứng dụng thương mại điện tử. Với những yêu cầu như trên, nhóm quyết định sử dụng công nghệ Winform C# cho việc làm giao diện, xử lý event và Microsoft SQL Server để xử lý dữ liệu.

## Dữ liệu, thông tin đầu vào yêu cầu

Khi sử dụng ứng dụng, người dùng được yêu cầu phải đăng ký một tài khoản bao gồm các thông tin cơ bản nhất như: họ tên người dùng, tên đăng nhập, mật khẩu, số điện thoại. Tuy nhiên, để có thể thực hiện đặt hàng và thanh toán, người dùng được yêu cầu phải cung cấp thêm các thông tin cần thiết như: thông tin về địa chỉ người dùng, thông tin thanh toán, quốc gia của người dùng,… Việc này trên thực tế giúp người dùng có thể tiện lợi trong việc đặt hàng, vừa giúp hệ thống trở nên dễ dàng hơn trong việc thao tác, thanh toán, giúp tiết kiệm thời gian của người dùng. Đối với người quản lý, họ sẽ được cung cấp một tài khoản để có thể thao tác mọi thứ trên ứng dụng.

## Mục đích, tính năng của ứng dụng

Đối với giao diện người dùng, ứng dụng sẽ mang đến các tính năng: hiển thị thông tin trực quan toàn bộ các sản phẩm đang được kinh doanh theo từng danh mục cụ thể với hình ảnh minh hoạ chi tiết; có đầy đủ các button để giúp người dùng có thể thêm sản phẩm vào giỏ hàng, chỉnh sửa số lượng cũng như có thể chọn thông tin thanh toán theo ý mình khi thanh toán; hiển thị toàn bộ các order, hoá đơn của người dùng, trạng thái của nó và chi tiết về sản phẩm trong order đó; cho phép người dùng chỉnh sửa các thông tin cá nhân, mật khẩu hoặc thêm, xoá các thông tin thanh toán của bản thân một cách trực quan nhất. Mục đích của việc phát triển giao diện người dùng là cung cấp một giao diện để người dùng có thể thao tác cơ bản, dễ dàng các tính năng của một ứng dụng thương mại điện tử. Giao diện cũng được xử lý tối ưu các phần load hình ảnh để có thể tăng trải nghiệm khi sử dụng, tránh việc load lại nhiều lần. Việc truy xuất database cũng được tối ưu hơn với việc sử dụng công nghệ Entity Framework.

Đối với giao diện của người quản lý, ứng dụng cũng mang đến các tính năng như sau: thông tin về tình hình kinh doanh của các mặt hàng; thêm, xoá, sửa thông tin của các sản phẩm có trong dữ liệu như tên sản phẩm, mô tả, ; thêm, xoá, sửa các đợt giảm giá để phục vụ cho việc kinh doanh tốt hơn; xem xét được toàn bộ thông tin của các hoá đơn, đặt hàng của người dùng, tiến hành hoàn thành thanh toán đơn hàng khi cần thiết; thêm, xoá, sửa các user trên ứng dụng theo nhu cầu. Việc cung cấp các tính năng nhằm mục đích giúp người quản lý có thể nắm được tình hình kinh doanh, đưa ra các giải pháp mang tính thương mại như thay đổi giá sản phẩm hay các đợt giảm giá cần thiết. Cũng như khi người dùng gặp khó khăn trong vấn đề đăng nhập, người quản lý có thể dễ dàng chỉnh sửa mật khẩu, cấp tài khoản mới,…

## Giao diện dự kiến

Khi tiến hành mở ứng dụng, người dùng sẽ tiến hành đăng nhập hoặc đăng ký tài khoản trên giao diện ứng dụng cung cấp để tiến hành vào bên trong ứng dụng. Việc hiển thị giao diện của người dùng hay của người quản lý sẽ dựa vào thông tin đăng nhập được cung cấp để phân loại

### Giao diện người dùng

Gồm thanh menu bên trái hiển thị toàn bộ các trang theo từng chức năng:

* Dashboard: Nơi hiển thị toàn bộ danh mục sản phẩm, thông tin sản phẩm chi tiết, hình ảnh và là form chính của đề tài. Người dùng có thể chọn danh mục để xem các sản phẩm trong đó. Mỗi sản phẩm sẽ có 2 button là “Add to Cart” và “Buy Now” thực hiện chức năng thêm sản phẩm vào giỏ hàng hoặc mua sản phẩm
* ShoppingCart: Nơi hiển thị những sản phẩm người dùng đã thêm vào giỏ hàng cũng như tổng tiền của các sản phẩm. Mỗi sản phẩm sẽ có nút tăng, giảm số lượng theo nhu cầu và giá tiền của sản phẩm đó. Giao diện có thể hiện tổng tiền của toàn bộ sản phẩm cũng như nút “Check Out” để người dùng tiến hành đặt hàng. Các phương thức thanh toán của người dùng sẽ được hiển thị để người dùng lựa chọn.
* MyOrders: Nơi hiển thị toàn bộ những đơn hàng người dùng đang đặt hàng và tình trạng thanh toán cũng như giao hàng của đơn hàng đó. Người dùng có thể xem chi tiết các sản phẩm có trong đơn hàng dựa trên button “Details”
* Invoices: Nơi hiển thị toàn bộ những hoá đơn của người dùng đã thanh toán và giao hàng thành công gồm các thông tin chi tiết về hoá đơn
* Account: Nơi hiển thị thông tin của người dùng được chia ra 3 phần là thông tin cơ bản, mật khẩu và thông tin thanh toán với 3 button “Information”, “Password” và “Payment”. Người dùng có thể thao tác để tiến hành thêm, xoá, sửa các thông tin này
* Sign Out: Đăng xuất khỏi ứng dụng và trở về giao diện đăng nhập

### Giao diện người quản lý

Gồm thanh menu bên trái hiển thị toàn bộ các trang theo từng chức năng:

* Dashboard: Hiển thị thông tin về tình trạng kinh doanh của ứng dụng với các thông tin về sản phẩm bán được, sản phẩm bán chạy, doanh thu,…
* All Products: Nơi hiển thị toàn bộ các sản phẩm có trong cơ sở dữ liệu với các thông tin chi tiết có thể thao tác. Người quản lý có thể chỉnh sửa thông tin sản phẩm, ẩn sản phẩm khỏi mục kinh doanh hoặc thêm sản phẩm mới với các button được hiển thị trên màn hình
* Discount: Hiển thị toàn bộ các đợt giảm giá mà người quản lý đã tạo. Người quản lý có thể tắt bật từng đợt giảm giá theo nhu cầu và có thể chọn sản phẩm cho đợt giảm giá đó. Người quản lý có thể tạo đợt giảm giá mới với button “Add Discount”
* Invoices: Hiển thị các đơn hàng người dùng đã đặt hàng hoặc đã thanh toán với các thông tin: ngày đặt hàng, ngày thanh toán, nhà cung cấp, trạng thái thanh toán. Mỗi đơn hàng có 1 button “Complete” để người quản lý xác nhận đơn hàng đã được thanh toán và giao hàng thành công
* User Accounts: Hiển thị các người dùng có trong dữ liệu với username và password. Mỗi người dùng sẽ có 2 button “Edit” và “Remove” thực hiện chức năng chỉnh sửa và xoá. Giao diện có button “Add new User” để tiến hành thêm người dùng mới vào hệ thống
* Sign Out: Đăng xuất khỏi ứng dụng và trở về giao diện đăng nhập

# CHƯƠNG 2: PHÂN CÔNG CÔNG VIỆC

##### Bảng 1: Phân công công việc

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Họ và tên | Mô tả công việc | % đóng góp |
| 1 | Nguyễn Gia Huy | Thiết kế, truy xuất cơ sở dữ liệu dựa trên Model; xử lý Event cho các Controller thao tác của người dùng và người quản lý; làm báo cáo | 100% |
| 2 | Huỳnh Lê Huy | Thiết kế giao diện View của người dùng; xử lý một số Event | 100% |
| 3 | Nguyễn Tường Phương Uyên | Thiết kế giao diện của form người quản lý; Làm slide | 100% |

# CHƯƠNG 3: THIẾT KẾ ỨNG DỤNG

## 3.1. Quá trình thiết kế

Đối với một ứng dụng form, nhóm đã sử dụng mô hình MVC để tiến hành xây dựng và code dựa trên những Model được thiết trong cơ sở dữ liệu và sự liên kết của các bảng. Sau đó, nhóm tiến hành đưa các Model này vào chương trình và tiến hành làm giao diện View cho từng chức năng. Nhóm sử dụng các class trong thư mục FormFunction để có thể truy xuất, xử lí dữ liệu theo các yêu cầu trong Controller của người dùng. Một View sẽ có nhiều Controller theo từng chức năng sẽ được xử lý và hiển thị.

Về ứng dụng thương mại điện tử, việc chia theo 2 View hoàn toàn độc lập là View người dùng và View người quản lý sẽ giúp dễ dàng hơn trong việc xử lý và thao tác. Việc phân công công việc dựa trên mô hình này cũng dễ dàng hơn trong quá trình làm việc của nhóm

## 3.2. Thiết kế lớp

##### Bảng 2A: Sinh viên phụ trách: Nguyễn Gia Huy (Các Class)

|  |  |  |
| --- | --- | --- |
| TT | Tên Class | Mục đích, chức năng |
| 1 | SessionFunction | Class chứa các chức năng liên quan đến việc sử dụng trong phiên của người dùng |
| 2 | UserFunction | Các chức năng liên quan đến việc hiển thị, thêm, xoá, sửa toàn bộ các thông tin của user |
| 3 | RegisterFunction | Chức năng đăng ký người dùng mới |
| 4 | ProductFunction | Toàn bộ các chức năng liên quan đến sản phẩm trong hệ thống |
| 5 | OrderFunction | Toàn bộ các chức năng liên quan đến đặt hàng và hoá đơn |
| 6 | LoginFunction | Chức năng đăng nhập của người dùng |
| 7 | DiscountFunction | Toàn bộ các chức năng thêm, xoá, sửa, trạng thái của giảm giá |
| 8 | CartFunction | Toàn bộ các chức năng thêm, xoá, sửa, của giỏ hàng người dùng |
| 9 | DrawCartController | Vẽ nên các panel trong phần giỏ hàng của người dùng |

##### Bảng 2B: Sinh viên phụ trách: Nguyễn Gia Huy (Các phương thức trong Class)

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Phương thức | Mục đích | Tên file, số thứ tự dòng |
| 1 | createSession(int user\_id)  Input: user\_id  Pseudo code:  Find (user\_id), total = 0, created\_at = DateTime.Now | Tạo phiên mới mỗi khi người dùng đăng nhập vào ứng dụng để tính toán tổng tiền trong giỏ hàng | SessionFunction.cs  Dòng: 13 |
| 2 | getIdCurrentSession(int user\_id)  input: user\_id  output: Id của phiên hiện tại | Dựa vào id của người dùng để lấy ra id của phiên hiện tại | SessionFunction.cs  Dòng: 27 |
| 3 | averageTotal()  output: decimal price | Tính trung bình tổng tiền người dùng sử dụng để thống kê | SessionFunction.cs  Dòng: 35 |
| 4 | displayUserInfo(int userId)  input: userId  output: Model<user> | Hiển thị thông tin cơ bản của một người dùng | UserFunction.cs  Dòng: 12 |
| 5 | listUser()  output: List<user> | Danh sách các user và thông tin của user | UserFunction.cs  Dòng: 20 |
| 6 | displayUserAddressInfo(int userId)  input: userId  output: Model<user\_address> | Hiển thị thông tin địa chỉ của một user | UserFunction.cs  Dòng: 27 |
| 7 | displayUserPaymentInfo(int userId)  input: userId  output: Model<user\_payment> | Hiển thị thông tin thanh toán của một user | UserFunction.cs  Dòng: 35 |
| 8 | removeUser(int userId)  input: userId | Xoá user khỏi hệ thống | UserFunction.cs  Dòng: 43 |
| 9 | alterUserPassword(int userId, String password)  input: userId, password | Đổi password của 1 user dựa vào id user | UserFunction.cs  Dòng: 52 |
| 10 | alterUserInfo(int userId, String firstName, String lastName, int telephone)  input: userId, firstName, lastName, telephone | Đổi thông tin cơ bản của 1 user dựa trên id user | UserFunction.cs  Dòng: 61 |
| 11 | alterUserAddressInfo(int userId, String address\_1, String address\_2, String city, String postal\_code, String country, String telephone, String mobile)  input: userId, address\_1, address\_2, city, postal\_code, country, telephone, mobile | Cập nhật lại toàn bộ thông tin về địa chỉ của người dùng | UserFunction.cs  Dòng: 72 |
| 12 | alterUserPaymentInfo(int id, String payment\_type, String providers, int account\_no, DateTime expiry)  input: id, payment\_type, providers, account\_no, expiry | Cập nhật lại thông tin về một phương thức thanh toán của người dùng | UserFunction.cs  Dòng: 87 |
| 13 | removeUserPaymentInfo(int id)  input: id | Xoá thông tin về một phương thức thanh toán của người dùng | UserFunction.cs  Dòng: 99 |
| 14 | totalUser() | Tổng người dùng có mặt trong hệ thống | UserFunction.cs  Dòng: 108 |
| 15 | addUserPaymentInfo(int userId, String payment\_type, String providers, int account\_no, DateTime expiry)  input: userId, payment\_type, providers, account\_no, expiry | Thêm thông tin về một phương thức thanh toán của người dùng | UserFunction.cs  Dòng: 115 |
| 16 | Register(String firstName, String lastName, String username, String passwords, int telephone)  Input: firstName, lastName, username, passwords, telephone | Đăng ký một người dùng mới vào hệ thống | RegisterFunction.cs  Dòng: 10 |
| 17 | listProduct(int category, int orderBy)  input: category, orderBy  output: List<dynamic>  pseudo code:  if (cate\_id == cate && !p.id.deleted\_at.HasValue)  orderBy(p.price,orderBy) | Lấy thông tin các sản phẩm theo danh mục và sắp xếp dựa trên orderBy | ProductFunction.cs  Dòng: 13 |
| 18 | listAllProducts()  output: List<product> | Thông tin tất cả các sản phẩm | ProductFunction.cs  Dòng: 26 |
| 19 | changeStateProduct(int product\_id, int state)  input: product\_id, state | Khiến cho 1 sản phẩm hiển thị hoặc không hiển thị với người dùng | ProductFunction.cs  Dòng: 33 |
| 20 | addProduct(string name\_product, string desc\_product, int category\_id, int quantity, decimal price, string image\_url)  input: name\_product, desc\_product, category\_id, quantity, price, image\_url  pseudo: addInven(quantity)  new product(input) | Thêm sản phẩm vào hệ thống với các thông tin | ProductFunction.cs  Dòng: 52 |
| 21 | alterProduct(int productId, string name\_product, string desc\_product,int inventory\_id, int quantity, decimal price, string image\_url)  input: productId ,name\_product, desc\_product, category\_id, quantity, price, image\_url | Chỉnh sửa thông tin của một sản phẩm trong hệ thống | ProductFunction.cs  Dòng: 77 |
| 22 | deleteProduct(int productID)  input: productId | Xoá một sản phẩm khỏi hệ thống | ProductFunction.cs  Dòng: 90 |
| 23 | detailProduct(int id)  Input: id  Output: List<dynamic> | Lấy thông tin cụ thể về một sản phẩm và liên quan | ProductFunction.cs  Dòng: 99 |
| 24 | categoryProduct()  output: List<product\_category> | Lấy thông tin về danh mục sản phẩm | ProductFunction.cs  Dòng: 106 |
| 25 | itemSoldQuantity()  output: int | Số lượng sản phẩm đã bán | ProductFunction.cs  Dòng: 114 |
| 26 | itemSoldToday()  output: int | Số lượng sản phẩm đã bán được trong hôm nay | ProductFunction.cs  Dòng: 122 |
| 27 | topSellingProduct()  output: string | Tên sản phẩm bán chạy nhất | ProductFunction.cs  Dòng: 132 |
| 28 | ordersInfo(int user\_id)  input: user\_id  output: List<dynamic> | Toàn bộ đơn đặt hàng nhưng chưa thanh toán của user | OrderFunction.cs  Dòng: 12 |
| 29 | List<dynamic> alls() | Toàn bộ đơn đặt hàng và hoá đơn của người dùng | OrderFunction.cs  Dòng: 20 |
| 30 | decimal totalMoneyEarned() | Tổng tiền đã thanh toán | OrderFunction.cs  Dòng: 28 |
| 31 | List<dynamic> invoicesInfo(int user\_id) | Toàn bộ đơn đặt hàng đã thanh toán của user | OrderFunction.cs  Dòng: 36 |
| 32 | int totalInvoices() | Tổng số đơn hàng đã thanh toán | OrderFunction.cs  Dòng: 44 |
| 33 | orderToInvoice(int idOrder) | Xử lý việc thanh toán hoàn tất của người dùng | OrderFunction.cs  Dòng: 52 |
| 34 | List<dynamic> orderItemsInfo(int order\_id) | Toàn bộ thông tin các sản phẩm có trong đơn đặt hàng | OrderFunction.cs  Dòng: 62 |
| 35 | void checkOut(int user\_id, int session\_id, int user\_payment\_id)  pseudo: new payment\_details(DateTime.Now, user\_id)  new order\_details(user\_id, total)  new order\_items(cart\_items) | Tiến hành đặt đơn hàng cho người dùng, tạo lần lượt 3 record vào 3 bảng trong csdl | OrderFunction.cs  Dòng: 70 |
| 36 | int checkLoginCredential(String username, String password) | Tiến hành check thông tin người dùng cung cấp trong dữ liệu | LoginFunction.cs  Dòng: 12 |
| 37 | List<discount> listDiscount() | Danh sách thông tin các đợt giảm giá | DiscountFunction.cs  Dòng: 12 |
| 38 | void addDiscount(string name, string desc, decimal percent, int state) | Thêm đợt giảm giá mới vào dữ liệu | DiscountFunction.cs  Dòng: 19 |
| 39 | void changeStateDiscount(int id, int state) | Bật tắt ảnh hưởng của đợt giảm giá | DiscountFunction.cs  Dòng: 35 |
| 40 | void alterProductToDiscount(int discountID, int productID, int turn) | Thêm hoặc xoá sản phẩm khỏi đợt giảm giá | DiscountFunction.cs  Dòng: 44 |
| 41 | void alterDiscount(int id, string name, string desc, decimal percent) | Chỉnh sửa thông tin của đợt giảm giá | DiscountFunction.cs  Dòng: 64 |
| 42 | List<dynamic> cartItemsInfo(int session\_id) | Lấy thông tin giỏ hàng đựa trên mã phiên | CartFunction.cs  Dòng: 14 |
| 43 | int quantityCart(int session\_id) | Tổng số lượng món hàng có trong đơn hàng | CartFunction.cs  Dòng: 47 |
| 44 | decimal totalCartInfo(int session\_id) | Tổng tiền của đơn hàng | CartFunction.cs  Dòng: 55 |
| 45 | void addToCart(int session\_id, int product\_id) | Thêm một sản phẩm vào giỏ hàng | CartFunction.cs  Dòng: 62 |
| 46 | void alterQuantityItem(int id, int quantity, int session\_id) | Chỉnh sửa số lượng của sản phẩm trong giỏ hàng | CartFunction.cs  Dòng: 93 |
| 47 | void removeItem(int id, int quantity, int session\_id) | Xoá một sản phẩm khỏi giỏ hàng | CartFunction.cs  Dòng: 109 |
| 48 | DrawCartController(string name, int price, int id, int count, MainForm parent) | Khởi tạo thông tin của các giỏ hàng cho các panel | DrawCartController.cs  Dòng: 18 |
| 49 | void Draw(Panel parent) | Vẽ nên các panel trong phần Shopping Cart | DrawCartController.cs  Dòng: 30 |
| 50 | void LbRemove\_Click(object sender, EventArgs e) | Xử lý phần xoá sản phẩm khỏi giỏ hàng người dùng | DrawCartController.cs  Dòng: 98 |
| 51 | void BtIncrease\_Click(object sender, EventArgs e) | Xử lý phần thêm số lượng sản phẩm khỏi giỏ hàng người dùng | DrawCartController.cs  Dòng: 104 |
| 52 | void BtDecrease\_Click(object sender, EventArgs e) | Xử lý phần giảm số lượng sản phẩm khỏi giỏ hàng người dùng | DrawCartController.cs  Dòng: 110 |

##### Bảng 3A: Sinh viên phụ trách: Huỳnh Lê Huy (Các Class)

|  |  |  |
| --- | --- | --- |
| TT | Tên Class | Mục đích, chức năng |
| 1 | DashboardController | Hiển thị giao diện trên menu Dashboard người dùng |
| 2 | InvoiceController | Hiển thị giao diện trên menu Invoices người dùng |
| 3 | MyOrderController | Hiển thị giao diện trên menu MyOrders người dùng |
| 4 | CartController | Hiển thị giao diện trên menu ShoppingCart người dùng |
| 5 | AcountController | Hiển thị giao diện trên menu Account người dùng |
| 6 | AboutUsController | Hiển thị giao diện AboutUs trên menu |
| 7 | DrawAccount | Vẽ nên giao diện Account của người dùng |
| 8 | DrawDeltail | Vẽ nên giao diện chi tiết các mặt hàng trong hoá đơn |
| 9 | DrawInvoice | Vẽ nên giao diện chi tiết các hoá đơn trong form người dùng |

##### Bảng 3B: Sinh viên phụ trách: Huỳnh Lê Huy (Các phương thức trong Class)

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Phương thức | Mục đích | Tên file, số thứ tự dòng |
| 1 | Void DashboardHeader() | Dựa vào các danh mục để vẽ lên các nút bấm theo từng danh mục sản phẩm | DashboardController.cs  Dòng: 46 |
| 2 | void DashboardContent() | Hiển thị giao diện các panel sản phẩm với các thông tin | DashboardController.cs  Dòng: 76 |
| 3 | void Lb\_Click(object sender, EventArgs e) | Xử lý event cho các button ở phần DashboardHeader() | DashboardController.cs  Dòng: 107 |
| 4 | DashboardController(Panel Dashboard, Panel pnHeader, Panel pnContent, int user\_id, MainForm parent) | Khởi tạo các giá trị giao diện dựa lựa chọn sản phẩm | DashboardController.cs  Dòng: 23 |
| 5 | InvoiceController(Panel Invoice, Panel pnHeader, Panel pnContent) | Khởi tạo panel chứa giao diện cho phần Invoices | InvoiceController.cs  Dòng: 13 |
| 6 | void InvoiceHeader() | Khởi tạo phần header của invoices | InvoiceController.cs  Dòng: 45 |
| 7 | void InvoiceTitle() | Khởi tạo các cột của bảng chứa các đơn hàng | InvoiceController.cs  Dòng: 57 |
| 8 | void InvoiceContent() | Gọi hàm vẽ nên các panel đặt hàng | InvoiceController.cs  Dòng: 91 |
| 9 | MyOrderController(Panel MyOrder, Panel pnHeader, Panel pnContent) | Khởi tạo các panel của phần MyOrder | MyOrderController.cs  Dòng: 23 |
| 10 | void MyOrderHeader() | Vẽ nên giao diện của Header | MyOrderController.cs  Dòng: 51 |
| 11 | void MyOrderTitle() | Vẽ nên phần các cột của MyOrder | MyOrderController.cs  Dòng: 63 |
| 12 | void MyOrderContent() | Gọi hàm để vẽ nên các panel chứa các hoá đơn | MyOrderController.cs  Dòng: 98 |
| 13 | CartController(Panel Cart, Panel pnHeader, Panel pnContent, int user\_id, MainForm parent) | Khởi tạo các panel của phần ShoppingCart | CartController.cs  Dòng: 25 |
| 14 | void CartHeader() | Vẽ nên giao diện của Header | CartController.cs  Dòng: 60 |
| 15 | void LbCheckOut\_Click(object sender, EventArgs e) | Xử lý event khi người dùng bấm thanh toán | CartController.cs  Dòng: 79 |
| 16 | void CartContent() | Gọi hàm để vẽ nên các panel chứa các sản phẩm trong giỏ hàng | CartController.cs  Dòng: 91 |
| 17 | void CartTotal() | Vẽ nên giao diện của phần tổng kết sản phẩm | CartController.cs  Dòng: 101 |
| 18 | AcountController(Panel Account, Panel pnHeader, Panel pnContent, MainForm parent) | Khởi tạo các panel của phần Account của người dùng | AcountController.cs  Dòng: 17 |
| 19 | GetDataForInformation() | Lấy dữ liệu của người dùng | AcountController.cs  Dòng: 24 |
| 20 | AboutUsController(Panel aboutUs, Panel pnHeader, Panel pnContent) | Khởi tạo các panel của phần AboutUs của người dùng | AboutUsController.cs  Dòng: 16 |
| 21 | void AboutHeader() | Vẽ nên giao diện của Header | AboutUsController.cs  Dòng: 35 |
| 22 | void AboutContent() | Vẽ nên giao diện của Content | AboutUsController.cs  Dòng: 47 |
| 23 | DrawAccount(Panel Account, Panel pnHeader, Panel pnContent, MainForm parent) | Khởi tạo các panel của phần DrawAccount của người dùng | DrawAccount.cs  Dòng: 34 |
| 24 | void AccountHeader() | Vẽ nên giao diện của Header của người dùng | DrawAccount.cs  Dòng: 61 |
| 25 | void LbPassword\_Click(object sender, EventArgs e) | Xử lý sự kiện Click cho Label Password để người dùng đổi mật khẩu | DrawAccount.cs  Dòng: 99 |
| 26 | void LbInformation\_Click(object sender, EventArgs e) | Xử lý sự kiện Click cho Label Info để người dùng thay đổi các thông tin | DrawAccount.cs  Dòng: 105 |
| 27 | void AccountContent() | Vẽ nên giao diện của Content của người dùng | DrawAccount.cs  Dòng: |
| 28 | void InFor(Panel parent) | Vẽ nên giao diện của Information của người dùng | DrawAccount.cs  Dòng: 132 |
| 29 | void CancelInfor\_Click(object sender, EventArgs e) | Xử lý sự kiện Click cho Label CancelInfor để người dùng xoá các thay đổi | DrawAccount.cs  Dòng: 225 |
| 30 | void EditInfor\_Click(object sender, EventArgs e) | Xử lý sự kiện Click cho Label EditInfor để người dùng mở bảng thay đổi | DrawAccount.cs  Dòng: 237 |
| 31 | void EditPayment\_Click(object sender, EventArgs e) | Xử lý sự kiện Click cho Label Edit Payment để người dùng mở bảng thay đổi | DrawAccount.cs  Dòng: 150 |
| 32 | void Ok\_Click(object sender, EventArgs e) | Xử lý sự kiện Click cho Button OK để người dùng lưu mật khẩu | DrawAccount.cs  Dòng: 366 |
| 33 | void Save\_Click(object sender, EventArgs e) | Xử lý sự kiện Click cho Button Save để người dùng lưu thông tin | DrawAccount.cs  Dòng: 402 |
| 34 | DrawDeltail(string name, int soLuong, decimal price) | Constructor nhận vào các thông tin chi tiết của sản phẩm | DrawDeltail.cs  Dòng: 16 |
| 35 | void Draw(Panel parent) | Vẽ nên các sản phẩm có trong hoá đơn người dùng | DrawDeltail.cs  Dòng: 35 |
| 36 | DrawInvoice(int index, string stt, string buyDate, string checkOutDate, string supplier, string status, string totalPrice) | Constructor nhận vào thông tin chi tiết về hoá đơn mà người dùng đã thanh toán | DrawInvoice.cs  Dòng: 19 |
| 37 | void Draw(Panel parent) | Vẽ nên các của hoá đơn người dùng | DrawInvoice.cs  Dòng: 30 |

##### Bảng 4A: Sinh viên phụ trách: Nguyễn Tường Phương Uyên (Các Class)

|  |  |  |
| --- | --- | --- |
| TT | Tên Class | Mục đích, chức năng |
| 1 | AllProductController | Hiển thị giao diện trên menu AllProduct người quản lý |
| 2 | DashboardManagerController | Hiển thị giao diện trên menu Dashboard người quản lý |
| 3 | DiscountController | Hiển thị giao diện trên menu Discount người quản lý |
| 4 | InvoicesController | Hiển thị giao diện trên menu Invoices người quản lý |
| 5 | UserAccountController | Hiển thị giao diện trên menu User Account người quản lý |
| 6 | DrawDiscount | Vẽ nên các panel trong phần giảm giá của người quản lý |
| 7 | DrawInvoices | Vẽ nên các panel trong phần toàn bộ đơn đặt hàng và hoá đơn của người quản lý |
| 8 | DrawProduct | Vẽ nên các panel trong phần tinh chỉnh sản phẩm của người quản lý |
| 9 | DrawUserAccount | Vẽ nên các panel trong phần tinh chỉnh các tài khoản người dùng của người quản lý |
| 10 | ListProdcutsInDiscount | Vẽ nên các panel trong phần các sản phẩm thuộc về giảm giá |

##### Bảng 4B: Sinh viên phụ trách: Nguyễn Tường Phương Uyên (Các phương thức trong Class)

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Phương thức | Mục đích | Tên file, số thứ tự dòng |
| 1 | AllProductController(Panel allProducts, Panel pnHeader, Panel pnContent, int user\_id, ManagementForm.ManagerForm parent) | Khởi tạo các panel của phần AllProductController của người dùng | AllProductController.cs  Dòng: 25 |
| 2 | void AllProductsHeader() | Vẽ nên giao diện của Header | AllProductController.cs  Dòng: 48 |
| 3 | void AddProduct\_Click(object sender, EventArgs e) | Xử lý event cho các button ở phần thêm sản phẩm | AllProductController.cs  Dòng: 62 |
| 4 | void AllProductsContent() | Vẽ nên giao diện của Content ở AllProduct | AllProductController.cs  Dòng: 68 |
| 5 | DashboardManagerController(Panel aboutUs, Panel pnHeader, Panel pnContent) | Khởi tạo các panel của phần DashboardManager của Admin | DashboardManagerController  Dòng: 18 |
| 6 | void DashboardHeader() | Vẽ nên giao diện của Header của Dashboard Admin | DashboardManagerController  Dòng: 37 |
| 7 | void DashboardContent() | Vẽ nên giao diện của Content của Dashboard Admin | DashboardManagerController  Dòng: 49 |
| 8 | DiscountController(Panel Discount, Panel pnHeader, Panel pnContent) | Khởi tạo các panel của phần Discount của Admin | DiscountController.cs  Dòng: 25 |
| 9 | void DiscountHeader() | Vẽ nên giao diện của Header của Discount Admin | DiscountController.cs  Dòng: 53 |
| 10 | void AddDiscount\_Click(object sender, EventArgs e) | Xử lý phần thêm vào phần Discount mới cho Admin | DiscountController.cs  Dòng: 76 |
| 11 | void DiscountTitle() | Vẽ nên giao diện của Title của Discount Admin | DiscountController.cs  Dòng: 84 |
| 12 | void DiscountContent() | Vẽ nên giao diện của Content của Discount Admin | DiscountController.cs  Dòng: 107 |
| 13 | InvoicesController(Panel Invoice, Panel pnHeader, Panel pnContent) | Khởi tạo các panel của phần Invoices của Admin | InvoicesController.cs  Dòng: 20 |
| 14 | void InvoiceHeader() | Vẽ nên giao diện của Header của Invoice Admin | InvoicesController.cs  Dòng: 46 |
| 15 | void InvoiceTitle() | Vẽ nên giao diện của Title của Invoice Admin | InvoicesController.cs  Dòng: 58 |
| 16 | void InvoiceContent() | Vẽ nên giao diện của Content của Invoice Admin | InvoicesController.cs  Dòng: 92 |
| 17 | UserAccountController(Panel Invoice, Panel pnHeader, Panel pnContent) | Khởi tạo các panel của phần User Account của Admin | UserAccountController.cs  Dòng: 20 |
| 18 | void UserAccountHeader() | Vẽ nên giao diện của Header của User Account Admin | UserAccountController.cs  Dòng: 46 |
| 19 | void UserAccountTitle() | Vẽ nên giao diện của Title của User Account Admin | UserAccountController.cs  Dòng: 67 |
| 20 | void UserAccountContent() | Vẽ nên giao diện của Content của User Account Admin | UserAccountController.cs  Dòng: 73 |
| 21 | DrawDiscount(string name, string desc, decimal? percent, int state, int idDiscount) | Khởi tạo các panel của phần vẽ Discount của Admin | DrawDiscount.cs  Dòng: 24 |
| 22 | void Draw(Panel parent) | Vẽ Discount của Admin | DrawDiscount.cs  Dòng: 33 |
| 23 | void Edit\_Click(object sender, EventArgs e) | Xử lý phần sửa vào phần Discount mới cho Admin | DrawDiscount.cs  Dòng: 99 |
| 24 | void Add\_Click(object sender, EventArgs e) | Xử lý phần thêm sản phẩm vào phần Discount cho Admin | DrawDiscount.cs  Dòng: 126 |
| 25 | void BtSate\_Click(object sender, EventArgs e) | Xử lý đổi trạng thái cho Discount cho Admin | DrawDiscount.cs  Dòng: 138 |
| 26 | DrawProduct(PictureBox itemPic, string tbName, string tbDesc, string tbCategory, string tbInvestory, string tbCost, string url, int State, int id,int invenID, ManagementForm.ManagerForm parent) | Khởi tạo các panel của phần vẽ các sản phẩm của Admin | DrawProduct.cs  Dòng: 24 |
| 27 | void DrawProducts(Panel parent, int index) | Hàm tiến hành vẽ các sản phẩm của Admin | DrawProduct.cs  Dòng: 39 |
| 28 | void BtSate\_Click(object sender, EventArgs e) | Xử lý sự kiện thay đổi trạng thái sản phẩm form Admin | DrawProduct.cs  Dòng: 122 |
| 29 | void BtEdit\_Click(object sender, EventArgs e) | Xử lý sự kiện sửa thông tin sản phẩm form Admin | DrawProduct.cs  Dòng: 143 |
| 30 | DrawUserAccount(int index, string stt, string username, string pass, int id) | Khởi tạo các panel của phần vẽ toàn bộ panel người dùng của Admin | DrawUserAccount.cs  Dòng: 21 |
| 31 | void Draw(Panel parent) | Hàm tiến hành vẽ các người dùng trong form của Admin | DrawUserAccount.cs  Dòng: 30 |
| 32 | void Remove\_Click(object sender, EventArgs e) | Hàm tiến hành xoá dữ liệu người dùng trong form Admin | DrawUserAccount.cs  Dòng: 89 |
| 33 | void Edit\_Click(object sender, EventArgs e) | Hàm tiến hành sửa dữ liệu người dùng trong form Admin | DrawUserAccount.cs  Dòng: 95 |
| 34 | ListProdcutsInDiscount(int proID, string name, string cost, int status, int discountID) | Khởi tạo panel của danh sách sản phẩm có trong Discount | ListProdcutsInDiscount.cs  Dòng: 22 |
| 35 | void Draw(Panel parent) | Vẽ danh sách sản phẩm có trong Discount | ListProdcutsInDiscount.cs  Dòng: 30 |
| 36 | void BtStatus\_Click(object sender, EventArgs e) | Xử lý việc đổi trạng thái của sản phẩm trong Discount | ListProdcutsInDiscount.cs  Dòng: 73 |

## 3.3. Thiết kế cơ sở dữ liệu

### Diagram, schematic Description automatically generated3.3.1. Mối quan hệ giữa các bảng

#### Hình 1: Sơ đồ cơ sở dữ liệu

### 3.3.2. Mô tả các bảng

##### Bảng 5: Bảng mô tả các bảng

|  |  |  |
| --- | --- | --- |
| STT | Tên bảng | Mục đích |
| 1 | Users | Thông tin về cơ bản chi tiết về người dùng |
| 2 | Product | Thông tin về chi tiết về sản phẩm |
| 3 | Product\_Category | Thông tin về các loại sản phẩm |
| 4 | Product\_Inventory | Thông tin về số lượng sản phẩm còn trong kho |
| 5 | User\_Address | Thông tin địa chỉ của người dùng |
| 6 | User\_Payment | Thông tin thanh toán của người dùng |
| 7 | Payment\_Details | Thông tin thanh toán chi tiết của đơn hàng |
| 8 | Order\_Items | Các sản phẩm trong đơn đặt hàng và hoá đơn của người dùng |
| 9 | Order\_Details | Thông tin chi tiết về đơn đặt hàng và hoá đơn |
| 10 | Shopping\_session | Chứa thông tin về phiên của một người dùng |
| 11 | Discount | Thông tin giảm giá các mặt hàng |
| 12 | Cart\_Item | Các sản phẩm trong giỏ hàng của người dùng |

## 3.4. Bảng mô tả các Fields

##### Bảng 6: Bảng mô tả table Users

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên trường | Kiểu dữ liệu | Mục đích |
| 1 | id | Int | Khoá chính bằng ID |
| 2 | Username | Nvarchar | Tên đăng nhập của người dùng |
| 3 | Passwords | Nvarchar | Mật khẩu của người dùng |
| 4 | First\_name | Nvarchar | Tên của người dùng |
| 5 | Last\_name | Nvarchar | Họ và tên lót của người dùng |
| 6 | Telephone | Int | Số điện thoại của người dùng |
| 7 | Created\_at | Date | Ngày tạo tài khoản |
| 8 | Modified\_at | Date | Ngày gần nhất chỉnh sửa thông tin |

##### Bảng 7: Bảng mô tả table Product

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên trường | Kiểu dữ liệu | Mục đích |
| 1 | id | Int | Khoá chính bằng ID |
| 2 | Name\_product | Nvarchar | Tên của sản phẩm |
| 3 | Desc\_product | Nvarchar | Phần mô tả của sản phẩm |
| 4 | Category\_id | Int | Khoá phụ trỏ tới loại sản phẩm |
| 5 | Inventory\_id | int | Khoá phụ trỏ tới kho hàng |
| 6 | Price | decimal | Giá tiền của sản phẩm |
| 7 | Discount\_id | Int | Khoá phụ trỏ tới giảm giá sản phẩm |
| 8 | Image\_url | Nvarchar | Hình ảnh mô tả sản phẩm |
| 9 | Created\_at | Date | Ngày tạo sản phẩm |
| 10 | Modified\_at | Date | Ngày gần nhất chỉnh sửa thông tin |
| 11 | Deleted\_at | Date | Ngày xoá sản phẩm |

##### Bảng 8: Bảng mô tả table Product\_category

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên trường | Kiểu dữ liệu | Mục đích |
| 1 | id | Int | Khoá chính bằng ID |
| 2 | Name\_category | Nvarchar | Tên loại sản phẩm |
| 3 | Desc\_category | Nvarchar | Phần mô tả của loại sản phẩm |
| 4 | Created\_at | Date | Ngày tạo loại sản phẩm |
| 5 | Modified\_at | Date | Ngày gần nhất chỉnh sửa thông tin |
| 6 | Deleted\_at | Date | Ngày xoá loại sản phẩm |

##### Bảng 9: Bảng mô tả table Product\_Inventory

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên trường | Kiểu dữ liệu | Mục đích |
| 1 | id | Int | Khoá chính bằng ID |
| 2 | Quantity | int | Số lượng còn lại của sản phẩm |
| 3 | Created\_at | Date | Ngày tạo sản phẩm |
| 4 | Modified\_at | Date | Ngày gần nhất chỉnh sửa thông tin |

##### Bảng 10: Bảng mô tả table User\_address

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên trường | Kiểu dữ liệu | Mục đích |
| 1 | id | Int | Khoá chính bằng ID |
| 2 | User\_id | Nvarchar | Khoá phụ trỏ tới Users |
| 3 | Address\_line1 | Nvarchar | Địa chỉ chính của người dùng |
| 4 | Address\_line2 | Nvarchar | Địa chỉ phụ của người dùng |
| 5 | City | Nvarchar | Thành phố của người dùng |
| 6 | Postal\_code | Nvarchar | Mã postal của người dùng |
| 7 | Country | Nvarchar | Quốc gia của người dùng |
| 8 | Telephone | Nvarchar | Số điện thoại của người dùng |
| 9 | Mobile | Nvarchar | Số điện thoại của người dùng |

##### Bảng 11: Bảng mô tả table User\_payment

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên trường | Kiểu dữ liệu | Mục đích |
| 1 | id | Int | Khoá chính bằng ID |
| 2 | User\_id | Int | Khoá phụ trỏ tới Users |
| 3 | Payment\_type | Nvarchar | Loại thanh toán |
| 4 | Providers | Nvarchar | Nhà cung cấp |
| 5 | Account\_no | Int | Mã thẻ thanh toán |
| 6 | Expiry | Date | Ngày hết hạn |

##### Bảng 12: Bảng mô tả table Payment\_details

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên trường | Kiểu dữ liệu | Mục đích |
| 1 | id | Int | Khoá chính bằng ID |
| 2 | Order\_id | Nvarchar | Khoá phụ trỏ tới order |
| 3 | amount | Int | Số lượngd |
| 4 | providers | Nvarchar | Nhà cung cấp |
| 5 | Statuss | Nvarchar | Tình trạng đơn hàng |
| 6 | Created\_at | Date | Ngày tạo thanh toán |
| 7 | Modified\_at | Date | Ngày gần nhất chỉnh sửa thông tin |

##### Bảng 13: Bảng mô tả table Order\_items

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên trường | Kiểu dữ liệu | Mục đích |
| 1 | id | Int | Khoá chính bằng ID |
| 2 | order\_id | int | Khoá phụ trỏ tới order |
| 3 | product\_id | int | Khoá phụ trỏ tới product |
| 4 | quanlity | int | Số lượng |
| 5 | Created\_at | Date | Ngày tạo sản phẩm trong order |
| 6 | Modified\_at | Date | Ngày gần nhất chỉnh sửa thông tin |

##### Bảng 14: Bảng mô tả table Order\_details

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên trường | Kiểu dữ liệu | Mục đích |
| 1 | id | Int | Khoá chính bằng ID |
| 2 | users\_id | int | Khoá phụ trỏ tới User |
| 3 | total | decimal | Tổng tiền của order |
| 4 | payment\_id | int | Khoá phụ trỏ tới Payment |
| 5 | Created\_at | Date | Ngày tạo order |
| 6 | Modified\_at | Date | Ngày gần nhất chỉnh sửa thông tin |

##### Bảng 15: Bảng mô tả table shopping\_session

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên trường | Kiểu dữ liệu | Mục đích |
| 1 | id | Int | Khoá chính bằng ID |
| 2 | users\_id | int | Khoá phụ trỏ tới User |
| 3 | total | decimal | Tổng tiền của phiên đó |
| 4 | Created\_at | Date | Ngày tạo phiên |
| 5 | Modified\_at | Date | Ngày gần nhất chỉnh sửa thông tin |

##### Bảng 16: Bảng mô tả table discount

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên trường | Kiểu dữ liệu | Mục đích |
| 1 | id | Int | Khoá chính bằng ID |
| 2 | name\_discount | Nvarchar | Tên của đợt giảm giá |
| 3 | desc\_discount | Nvarchar | Phần mô tả của đợt giảm giá |
| 4 | discount\_percent | decimal | Phần trăm giảm giá |
| 5 | active | bit | Kích hoạt hay chưa |
| 6 | Created\_at | Date | Ngày tạo đợt giảm giá |
| 7 | Modified\_at | Date | Ngày gần nhất chỉnh sửa thông tin |
| 8 | Deleted\_at | Date | Ngày xoá đợt giảm giá |

##### Bảng 17: Bảng mô tả table cart\_item

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên trường | Kiểu dữ liệu | Mục đích |
| 1 | id | Int | Khoá chính bằng ID |
| 2 | session\_id | int | Khoá phụ trỏ tới Session |
| 3 | product\_id | int | Khoá phụ trỏ tới Product |
| 4 | quanlity | int | Số lượng sản phẩm |
| 5 | Created\_at | Date | Ngày tạo order |
| 6 | Modified\_at | Date | Ngày gần nhất chỉnh sửa thông tin |

## 3.5. Thiết kế giao diện

##### Bảng 18: Bảng tổng hợp thiết kế giao diện người dùng

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Màn hình/Cửa sổ/Dialog | Mục đích | Giải thích |
| 1 | Hình 2.1: Màn hình đăng nhập | Tạo giao diện đăng nhập trực quan, dễ dàng thao tác | Có chức năng để người dùng đăng nhập hoặc ấn vào button đăng ký. Button Sign in để tiến hành đăng nhập |
| 2 | Hình 2.2: Màn hình đăng ký | Giao diện đăng ký gần giống với giao diện đăng nhập, dễ dùng | Người dùng nhập vào các thông tin cần thiết sau đó bấm Sign Up để đăng ký. Bấm “already …” để về màn Sign In |
| 3 | Hình 2.3: Màn hình Dashboard người dùng | Giao diện giúp người dùng có được thông tin về sản phẩm, hình ảnh và có thể mua hoặc thêm vào giỏ hàng | Header là các danh mục sản phẩm để lựa chọn. Phần dưới là các sản phẩm có trong danh mục. 2 Button làm nhiệm vụ xử lý |
| 4 | Hình 2.4: Màn hình Shopping Cart người dùng | Giao diện giúp người dùng thấy được các mặt hàng trong giỏ, điều chỉnh và thanh toán | Gồm 2 button thêm và giảm số lượng, 1 button xoá sản phẩm khỏi giỏ hàng và 1 button Check Out |
| 5 | Hình 2.5: Giao diện chọn phương thức thanh toán | Giúp người dùng xem và chọn phương thức thanh toán cho đơn hàng | Gồm 1 button cho mỗi phương thức thanh toán giúp chọn dễ dàng |
| 6 | Hình 2.6: Giao diện hiển thị đơn đặt hàng chưa thanh toán | Giúp người dùng dễ dàng theo dõi các đơn đặt hàng của mình và các thông tin | Mỗi đơn sẽ có 1 button giúp xem chi tiết các sản phẩm có trong đơn hàng |
| 7 | Hình 2.7: Giao diện chi tiết sản phẩm trong đơn hàng | Giúp người dùng xem được sản phẩm có trong đơn hàng hiện tại | Dễ dàng theo dõi và cụ thể đơn hàng hơn |
| 8 | Hình 2.8: Giao diện hoá đơn người dùng đã thanh toán | Giúp người dùng theo dõi các hoá đơn và nhà cung cấp thanh toán | Giúp người dùng có thể thống kê được những lần thanh toán |
| 9 | Hình 2.9: Giao diện thông tin cơ bản người dùng | Giúp người dùng quan sát, thay đổi thông tin theo nhu cầu | Gồm có 1 button Edit khi ấn sẽ hiện ra button Save và Cancle. |
| 10 | Hình 2.10: Giao diện thay đổi mật khẩu | Giúp người dùng thay đổi mật khẩu theo ý muốn | Button OK để xác nhận thay đổi mật khẩu. Nếu có lỗi sẽ hiện lên thông báo |
| 11 | Hình 2.11: Giao diện thông tin thanh toán | Giúp người dùng thêm, xoá, sửa các thông tin thanh toán của mình | Button “add” giúp thêm thông tin thanh toán. “Edit” và “Remove” giúp tinh chỉnh |
| 12 | Hình 2.12: Giao diện thêm thông tin thanh toán | Giúp người dùng thêm vào thông tin thanh toán mới | Button “Save” để lưu lại dữ liệu |

##### Bảng 19: Bảng tổng hợp giao diện người quản lý

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Màn hình/Cửa sổ/Dialog | Mục đích | Giải thích |
| 1 | Hình 3.1: Giao diện Dashboard người quản lý | Giúp người quản lý có thể biết được tình hình kinh doanh của nền tảng | Đưa ra các thông tin như số lượng sản phẩm đã bán, sản phẩm bán chạy nhất,… |
| 2 | Hình 3.2: Giao diện All Products người quản lý | Giúp người quản lý chỉnh sửa các thông tin của sản phẩm theo nhu cầu | Gồm 3 button là “State” giúp tắt bật sản phẩm, “Edit” để sửa và “add new” để thêm mới |
| 3 | Hình 3.3: Giao diện thêm sản phẩm người quản lý | Giao diện giúp người dùng đưa các thông tin của sản phẩm mới vào dữ liệu | Gồm các ô giúp người dùng nhập liệu và 2 button để huỷ “Cancel” hoặc lưu “Save” |
| 4 | Hình 3.4: Giao diện Discount người quản lý | Giúp người quản lý có thể quan sát các đợt giảm giá và chỉnh sửa theo nhu cầu | Mỗi đợt giảm giá sẽ có 3 button: trạng thái “State”, “Edit” chỉnh sửa hoặc thêm sản phẩm mới vào đợt giảm giá |
| 5 | Hình 3.5: Giao diện thêm Discount | Giúp người dùng có thể thêm các đợt giảm giá mới vào hệ thống | Giao diện gồm các ô thông tin và 2 button “status” và “Save” |
| 6 | Hình 3.6: Giao diện Invoices người quản lý | Giao diện giúp quản lý có thể quan sát và hoàn thành đơn giao dịch của người dùng | Giao diện gồm các thông tin và button “Complete”giúp hoàn thành đơn hàng |
| 7 | Hình 3.7: Giao diện User Accounts người quản lý | Giao diện giúp người quản lý quan sát tài khoản người dùng và tiến hành chỉnh sủa theo nhu cầu | Gồm 2 button cho mỗi người dùng là “Edit” và “Remove”  Giúp thao tác dễ dàng |

# CHƯƠNG 4: CÀI ĐẶT VÀ KIỂM THỬ

##### Bảng 20: Bảng tình huống kiểm thử phần mềm

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Tình huống | Mục đích | Giải thích |
| 1 | Hình 4.1: Đăng ký người dùng mới  Hình 4.2: Xem thử thông tin người dùng  Hình 4.3: Thêm một số thông tin cơ bản  Hình 4.4: Thêm thông tin thanh toán | Kiểm thử phần đăng ký người dùng mới, tiến hành thêm thông tin cơ bản và thông tin thanh toán cho người dùng | Kiểm tra phần đăng ký và thêm các thông tin xem có hoạt động đúng theo yêu cầu chưa |
| 2 | Hình 4.5: Tiến hành thêm vài sản phẩm vào giỏ hàng  Hình 4.6: Tiến hành thanh toán bằng thẻ đã thêm  Hình 4.7: Đơn hàng đã được đặt thành công  Hình 4.8: Tiến hành vào Form Admin Complete đơn hàng  Hình 4.9: Hoá đơn đã được thanh toán thành công | Kiểm tra xem hệ thống giao dịch như thêm vào giỏ hàng, tiến hành thanh toán có hoạt động tốt không. Phần phê duyệt đơn hàng ở Admin và phần hiển thị lại sau khi thêm ở người dùng | Tiến hành thực hiện thêm hàng vào giỏ. Sau đó chọn thẻ thanh toán và kiểm tra đơn hàng. Người quản lý phê duyệt sau đó đơn hàng đã được thanh toán |
| 3 | Hình 4.10: Thông tin thống kê của người quản lý  Hình 4.10: Thông tin thống kê về hoá đơn | Kiểm tra xem các đơn hàng đã thanh toán có được thống kê một cách đầy đủ và tính toán chính xác không | Tiến hành xem xét các thông tin về thống kê và hoá đơn ở giao diện người quản lý |
| 4 | Hình 4.11: Thêm một discount mới vào hệ thống  Hình 4.12: Thêm 3 sản phẩm vào discount và bật lên  Hình 4.13: Giá sản phẩm được thêm đã thay đổi  Hình 4.14: Tính toán phần discount trong giỏ hàng | Kiểm tra xem hệ thống giảm giá có hoạt động chính xác trong trường hợp được thêm. Kiểm tra xem giá đã giảm khi bỏ vào giỏ hàng có được cập nhật và tính toán không | Tiến hành thêm đợt giảm giá giáng sinh 20% cho 3 sản phẩm trên hình. 3 sản phẩm đã được giảm đúng theo nhu cầu kể cả khi thêm vào giỏ hàng và tính toán total |
| 5 | Hình 4.15: Danh sách các người dùng trong hệ thống  Hình 4.16: Thay đổi mật khẩu cho người dùng | Kiểm tra hệ thống danh sách người dùng trong hệ thống và tiến hành đổi mật khẩu 1 user | Trong trường hợp muốn xem toàn bộ các user trong hệ thống và phải cập nhật mật khẩu cho người dùng. Admin có thể dùng form này |

# CHƯƠNG 5: KẾT LUẬN

## 5.1. Mức độ hoàn thành

Đối với đồ án môn học, ứng dụng đã có thể sử dụng, tận dụng đầy đủ các bảng trong cơ sở dữ liệu theo ý tưởng. Tuy nhiên, khi so với các ứng dụng trong thực tế, đồ án còn khá nhiều thiếu xót và chưa được tối ưu, bảo mật tốt. Đồ án đã hoàn thành được các chức năng khá cơ bản của một ứng dụng thương mại điện tử thường gặp.

## 5.2. Các khó khăn và giải pháp

Khó khăn:

* Việc chọn sử dụng Winform khiến cho việc lấy dữ liệu lên và gán sự kiện cho các chức năng trở lên khá khó khăn.
* Việc sắp xếp vị trí cho các thành phần trong ứng dụng cũng rất mất thời gian do công nghệ đã cũ
* Winform cũng làm chậm đi độ trải nghiệm của ứng dụng
* Việc truyền dữ liệu giữa các class là khá mất thời gian và không tối ưu, khiến cho code phần nào khó để nâng cấp về sau

Giải pháp:

* Sử dụng các công nghệ mới như các framework để có thể giúp dữ liệu được truy xuất nhanh hơn và dễ dàng hơn
* Tạo các biến static để không phải truyền nhiều dữ liệu giữa các class
* Viết các hàm chung để có thể dễ sử dụng hơn

## 5.3. Ưu điểm và hạn chế

Ưu điểm:

* Sử dụng MVC giúp cho việc code nhóm và phân chia công việc trở nên dễ dàng hơn
* Winform cung cấp những giao diện có sẵn để có thể thao tác
* SQL Server kết hợp với C# giúp dữ liệu được lấy tương đối nhanh chóng
* Ứng dụng cung cấp được những tính năng cơ bản
* Ứng dụng có giao diện trực quan đối với người dùng, dễ thao tác, chỉnh sửa

Hạn chế:

* Chuyển dữ liệu giữa các form khá khó khăn
* Một số tính năng chưa thực sự tối ưu
* Thiếu một số chức năng khi so với các ứng dụng

## 5.4. Ý tưởng phát triển đồ án

Như đã nói, đồ án sử dụng công nghệ C# Winform kết hợp với Entity Framework. Đây là các công nghệ đã khá cũ và khó để lập trình cũng như tối ưu hoá về sau. Để phát triển, nhóm có ý tưởng dùng một công nghệ làm ứng dụng mới hơn cũng dùng C# đó chính là Windows Presentation Foundation (WPF). Dựa trên các ý tưởng về cơ sở dữ liệu và cách chương trình hoạt động, WPF sẽ cung cấp một nền tảng tốt hơn và tối ưu hơn. Về mặt chức năng, ứng dụng còn khá nhiều thiếu xót và có thể cải thiện tốt hơn và thêm các chức năng khác trong tương lai để gần với thực tế hơn.