

WEBSITE QUẢN LÝ BÁN ÁO

“YOUNG SG”

Chuyên ngành：Ứng dụng phần mềm.

Lớp：IT16302

Nhóm：1

GVHD：thầy Lê Anh Tú

Thời gian：18/06/2022

**Tổng quan sản phẩm:**

“YOUNG SG” là một website chuyên về bán áo. Giao diện thiết kế theo hướng tối giản, dễ tiếp cận và dễ sử dụng. Với các chức năng hỗ trợ việc quản lý sản phẩm, trưng bày và bán sản phẩm đến khách hàng. Quản lý thông tin khách hàng. Chức năng thống kê sẽ lưu trữ lịch sử thanh toán và hóa đơn.



Logo của website.

**Danh sách thành viên thực hiện dự án:**

|  |  |  |
| --- | --- | --- |
| **Họ & tên** | **MSSV** | **Vai trò** |
| Lê Công Thành | PS15305 | Trưởng nhóm |
| Lê Thanh Tùng | PS15314 | Thành viên |
| Lê Hồng Quân | PK01553 | Thành viên |
| Đỗ Nguyễn Tấn Phát | PD05082 | Thành viên |
| Huỳnh Tấn Phát | PS15403 | Thành viên |
| Phan Hoài Bảo | PS15317 | Thành viên |

1. **YÊU CẦU:**
2. **Yêu cầu nghiệp vụ phía khách hàng:**
   1. Đăng nhập:
      1. Tài khoản của khách hàng:
         1. Tên tài khoản
         2. Mật khẩu
         3. Họ tên khách hàng
         4. Email
         5. Hình ảnh
         6. Trạng thái kích hoạt
      2. Tài khoản của người quản lý:
         1. Tên tài khoản
         2. Mật khẩu
         3. Họ tên khách hàng
         4. Email
         5. Hình ảnh
         6. Trạng thái kích hoạt
         7. Vai trò quản lý
   2. Đăng ký:
      1. Tài khoản khách hàng:
         1. Tên tài khoản
         2. Email
         3. Mật khẩu
      2. Tài khoản quản lý:
         1. Tên tài khoản
         2. Email
         3. Mật khẩu
         4. Vai trò quản lý
   3. Quên mật khẩu:
   4. Giỏ hàng:
      1. Thêm sản phẩm vào giỏ hàng
      2. Xóa sản phẩm khỏi giỏ hàng
      3. Tăng số lượng sản phẩm trong giỏ hàng
   5. Thanh toán:
      1. Địa chỉ
      2. Số điện thoại
   6. Lịch sử thanh toán / hóa đơn:
      1. Hóa đơn
      2. Hóa đơn chi tiết
3. **Yêu cầu nghiệp vụ phía người quản lý:**
   1. Quản lý sản phẩm:
      1. Thêm sản phẩm
      2. Xóa sản phẩm
      3. Cập nhật sản phẩm
      4. Xem chi tiết sản phẩm
   2. Quản lý tài khoản:
      1. Thêm tài khoản
      2. Xóa tài khoản
      3. Cập nhật tài khoản
      4. Xem chi tiết thông tin tài khoản
   3. Thống kê hóa đơn, lịch sử thanh toán:
      1. Xem chi tiết hóa đơn.
      2. Danh sách hóa đơn theo thời gian.
4. **Yêu cầu bảo mật:**

*\* Cho người dùng truy cập trong các trường hợp được liệt kê bên dưới*

|  |  |  |  |
| --- | --- | --- | --- |
| **MỤC** | **Chưa đăng nhập** | **Đã đăng nhập  (chưa xác thực)** | **Đã đăng nhập (đã xác thực)** |
| Trang chủ | ✓ | ✓ | ✓ |
| Quên mật khẩu | ✓ | ✗ | ✗ |
| Giới thiệu | ✓ | ✓ | ✓ |
| Truyền thông | ✓ | ✓ | ✓ |
| Blog | ✓ | ✓ | ✓ |
| Trợ giúp | ✓ | ✓ | ✓ |
| Quy định | ✓ | ✓ | ✓ |
| Quyền riêng tư | ✓ | ✓ | ✓ |
| Liên hệ hổ trợ | ✓ | ✓ | ✓ |
| Thông tin cá nhân | ✗ | ✓ | ✓ |
| Tìm kiếm | ✓ | ✓ | ✓ |
| Thêm sản phẩm | ✗ | ✗ | ✓ |
| Quản lý sản phẩm | ✗ | ✗ | ✓ |
| Đơn bán | ✗ | ✗ | ✓ |
| Đơn mua | ✗ | ✗ | ✓ |
| Sản phẩm thích | ✓ | ✓ | ✓ |
| Lịch sử giao dịch | ✗ | ✓ | ✓ |
| Giỏ hàng | ✗ | ✗ | ✓ |
| Thanh toán | ✗ | ✗ | ✓ |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

1. **Yêu cầu công nghệ:**
   1. Frontend
      1. Angular

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Angular hỗ trợ hầu hết các trình duyệt gần đây. Điều này bao gồm các phiên bản cụ thể sau:   |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | | Chrome | Firefox | Edge | IE | Safari | iOS | Android | IE mobile | | latest | latest | 14 | 11 | 10 | 10 | Marshmallow (6.0) | 11 | |  |  | 13 | 10 | 9 | 9 | Lollipop  (5.0, 5.1) |  | |  |  |  | 9 | 8 | 8 | KitKat  (4.4) |  | |  |  |  |  | 7 | 7 | Jelly Bean  (4.1, 4.2, 4.3) |  | |

* 1. Backend
     1. Eclipse for Java EE
     2. Srpingboot
     3. JDK 8

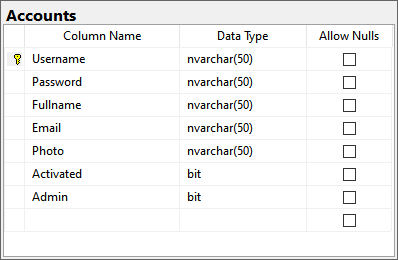
1. **THIẾT KẾ:**
2. **Thiết kế cơ sở dữ liệu:**
   1. **Các bảng dữ liệu:**

* **Account**
* Code

Tạo bảng:

1. **CREATE** **TABLE** [dbo].[Accounts](
2. [Username] [nvarchar](50) NOT NULL,
3. [**Password**] [nvarchar](50) NOT NULL,
4. [Fullname] [nvarchar](50) NOT NULL,
5. [Email] [nvarchar](50) NOT NULL,
6. [Photo] [nvarchar](50) NOT NULL,
7. [Activated] [**bit**] NOT NULL,
8. [Admin] [**bit**] NOT NULL,
9. **PRIMARY** **KEY** ([Username]))

* Bảng



* **Products**
* Code

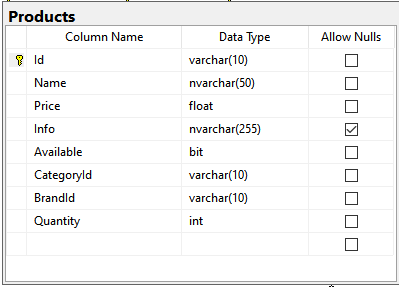
Tạo bảng:

1. **CREATE** **TABLE** [dbo].[Products](
2. [Id] [**varchar**](10) NOT NULL,
3. [**Name**] [nvarchar](50) NOT NULL,
4. [Price] [**float**] NOT NULL,
5. [Info] [nvarchar](255),
6. [Available] [**bit**] NOT NULL,
7. [CategoryId] [**varchar**](10) NOT NULL,
8. [BrandId] [**varchar**](10) NOT NULL,
9. [Quantity] [**int**] NOT NULL,
10. **PRIMARY** **KEY** ([Id])
11. )

Liên kết:

1. **ALTER** **TABLE** [dbo].[Products]
2. **ADD** **CONSTRAINT** FK\_Product\_Brand
3. **FOREIGN** **KEY**([BrandId]) **REFERENCES** [dbo].[Brand]([Id]);
4. /\***ON** **UPDATE** **CASCADE** **ON** **DELETE** **CASCADE**;    \*/
5. GO
6. **ALTER** **TABLE** [dbo].[Products]
7. **ADD** **CONSTRAINT** FK\_Product\_Category
8. **FOREIGN** **KEY**([CategoryId]) **REFERENCES** [dbo].[Category]([Id]);
9. /\***ON** **UPDATE** **CASCADE** **ON** **DELETE** **CASCADE**;    \*/

* Bảng

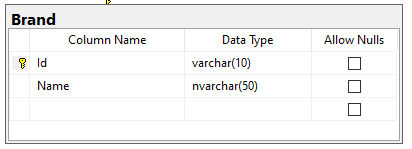


* **Brands**
* Code

Tạo bảng:

1. **CREATE** **TABLE** [dbo].[Brand](
2. [Id] [**varchar**](10) NOT NULL,
3. [**Name**] [nvarchar](50) NOT NULL,
4. **PRIMARY** **KEY** (Id))

* Bảng

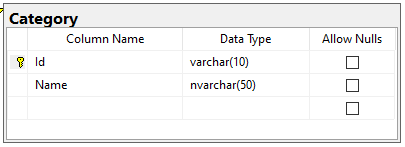


* **Category**
* Code

Tạo bảng:

1. **CREATE** **TABLE** [dbo].[Category](
2. [Id] [**varchar**](10) NOT NULL,
3. [**Name**] [nvarchar](50) NOT NULL,
4. **PRIMARY** **KEY** ([Id]))

* Bảng



* **Color**
* Code

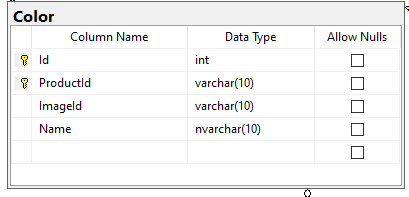
Tạo bảng:

1. **CREATE** **TABLE** [dbo].[Color](
2. [Id] [**int**] IDENTITY(1,1) NOT NULL,
3. [ProductId] [**varchar**](10) NOT NULL,
4. [ImageId] [**varchar**](10) NOT NULL,
5. [**Name**] [nvarchar](10) NOT NULL,
6. **PRIMARY** **KEY** ([Id],[ProductId]))

Liên kết:

1. GO
2. **ALTER** **TABLE** [dbo].[Color]
3. **ADD** **CONSTRAINT** FK\_Color\_Product
4. **FOREIGN** **KEY**([ProductId]) **REFERENCES** [dbo].[Products]([Id])
5. **ON** **UPDATE** **CASCADE** **ON** **DELETE** **CASCADE**;
6. GO
7. **ALTER** **TABLE** [dbo].[Color]
8. **ADD** **CONSTRAINT** FK\_Color\_Image
9. **FOREIGN** **KEY**([ImageId]) **REFERENCES** [dbo].[Image]([Id])
10. **ON** **UPDATE** **CASCADE** **ON** **DELETE** **CASCADE**;

* Bảng

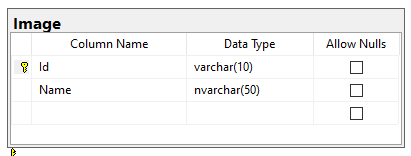


* **Images**
* Code

Tạo bảng:

1. **CREATE** **TABLE** [dbo].[Image](
2. [Id] [**varchar**](10) NOT NULL,
3. [**Name**] [nvarchar](50) NOT NULL,
4. **PRIMARY** **KEY** ([Id]))

* Bảng



* **Size**
* Code

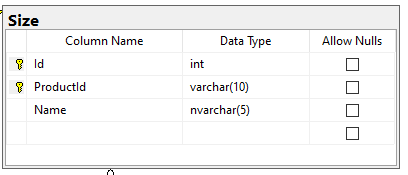
Tạo bảng:

1. **CREATE** **TABLE** [dbo].[**Size**](
2. [Id] [**int**] IDENTITY(1,1) NOT NULL,
3. [ProductId] [**varchar**](10) NOT NULL,
4. [**Name**] [nvarchar](5) NOT NULL,
5. **PRIMARY** **KEY** ([Id],[ProductId]))

Liên kết:

1. **ALTER** **TABLE** [dbo].[**Size**] **ADD** **CONSTRAINT** FK\_Size\_Product
2. **FOREIGN** **KEY**([ProductId]) **REFERENCES** [dbo].[Products]([Id])
3. **ON** **UPDATE** **CASCADE** **ON** **DELETE** **CASCADE**;

* Bảng



* **Orders**
* Code

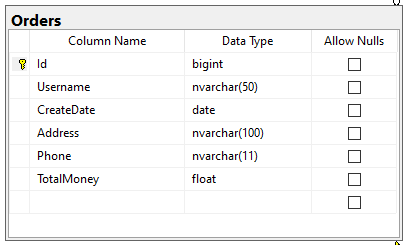
Tạo bảng:

1. **CREATE** **TABLE** [dbo].[Orders](
2. [Id] [**bigint**] IDENTITY(1,1) NOT NULL,
3. [Username] [nvarchar](50) NOT NULL,
4. [CreateDate] [**date**] NOT NULL,
5. [Address] [nvarchar](100) NOT NULL,
6. [Phone] [nvarchar](11) NOT NULL,
7. [TotalMoney]  [**float**] NOT NULL,
8. **PRIMARY** **KEY** ([Id]))

Liên kết:

1. GO
2. **ALTER** **TABLE** [dbo].[Orders]  **ADD** **CONSTRAINT** FK\_Orders\_Accounts
3. **FOREIGN** **KEY**([Username]) **REFERENCES** [dbo].[Accounts] ([Username])

* Bảng



* **OrderDetails**
* Code

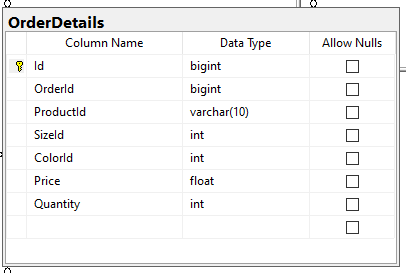
Tạo bảng:

1. **CREATE** **TABLE** [dbo].[OrderDetails](
2. [Id] [**bigint**] IDENTITY(1,1) NOT NULL,
3. [OrderId] [**bigint**] NOT NULL,
4. [ProductId] [**varchar**](10) NOT NULL,
5. [SizeId] [**int**] NOT NULL,
6. [ColorId] [**int**] NOT NULL,
7. [Price] [**float**] NOT NULL,
8. [Quantity] [**int**] NOT NULL,
9. **PRIMARY** **KEY** ([Id]))

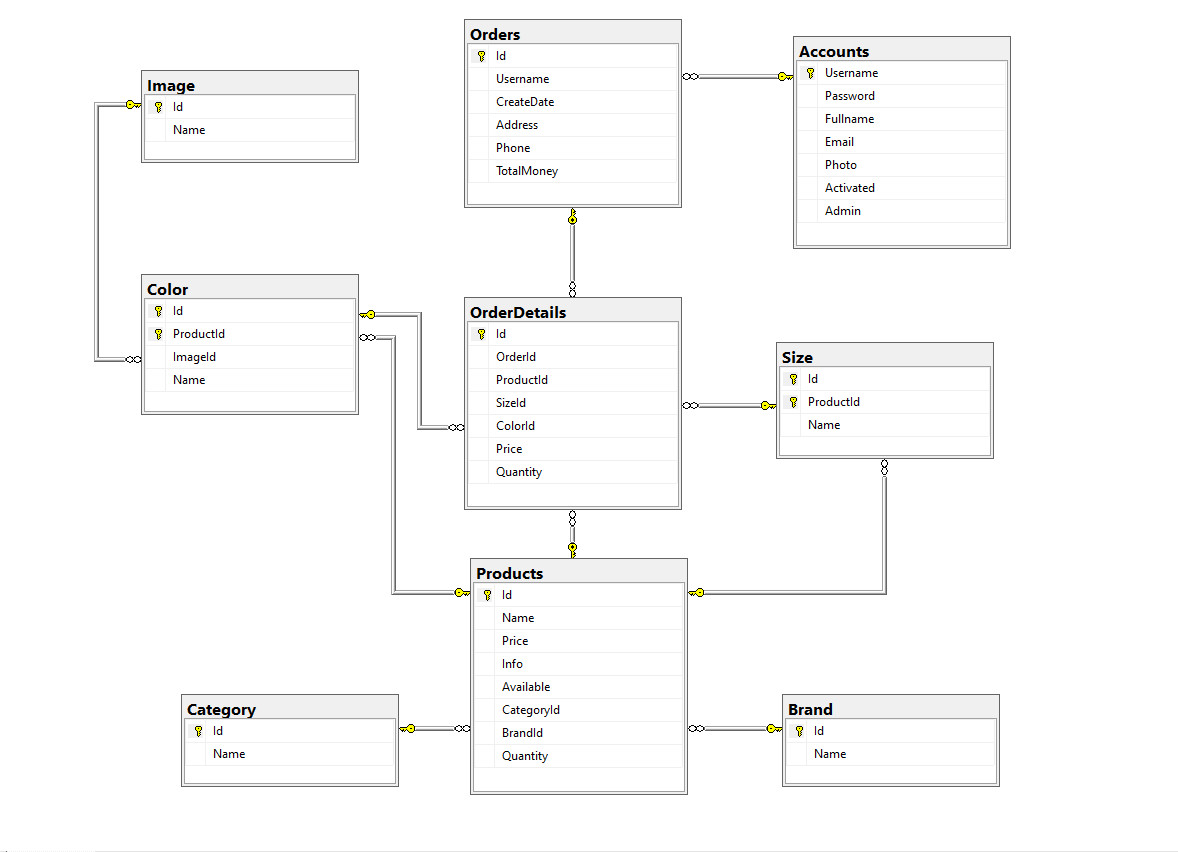
Liên kết:

1. **ALTER** **TABLE** [dbo].[OrderDetails]  **ADD** **CONSTRAINT** FK\_OrderDetails\_Orders
2. **FOREIGN** **KEY**([OrderId]) **REFERENCES** [dbo].[Orders] ([Id])
3. GO
4. **ALTER** **TABLE** [dbo].[OrderDetails]  **ADD** **CONSTRAINT** FK\_OrderDetails\_Product
5. **FOREIGN** **KEY**([ProductId]) **REFERENCES** [dbo].[Products] ([Id])
6. GO
7. **ALTER** **TABLE** [dbo].[OrderDetails]  **ADD** **CONSTRAINT** FK\_OrderDetails\_Size
8. **FOREIGN** **KEY**([SizeId],[ProductId]) **REFERENCES** [dbo].[**Size**] ([Id],[ProductId])
9. GO
10. **ALTER** **TABLE** [dbo].[OrderDetails]  **ADD** **CONSTRAINT** FK\_OrderDetails\_Color
11. **FOREIGN** **KEY**([ColorId],[ProductId]) **REFERENCES** [dbo].[Color] ([Id],[ProductId])

* Bảng

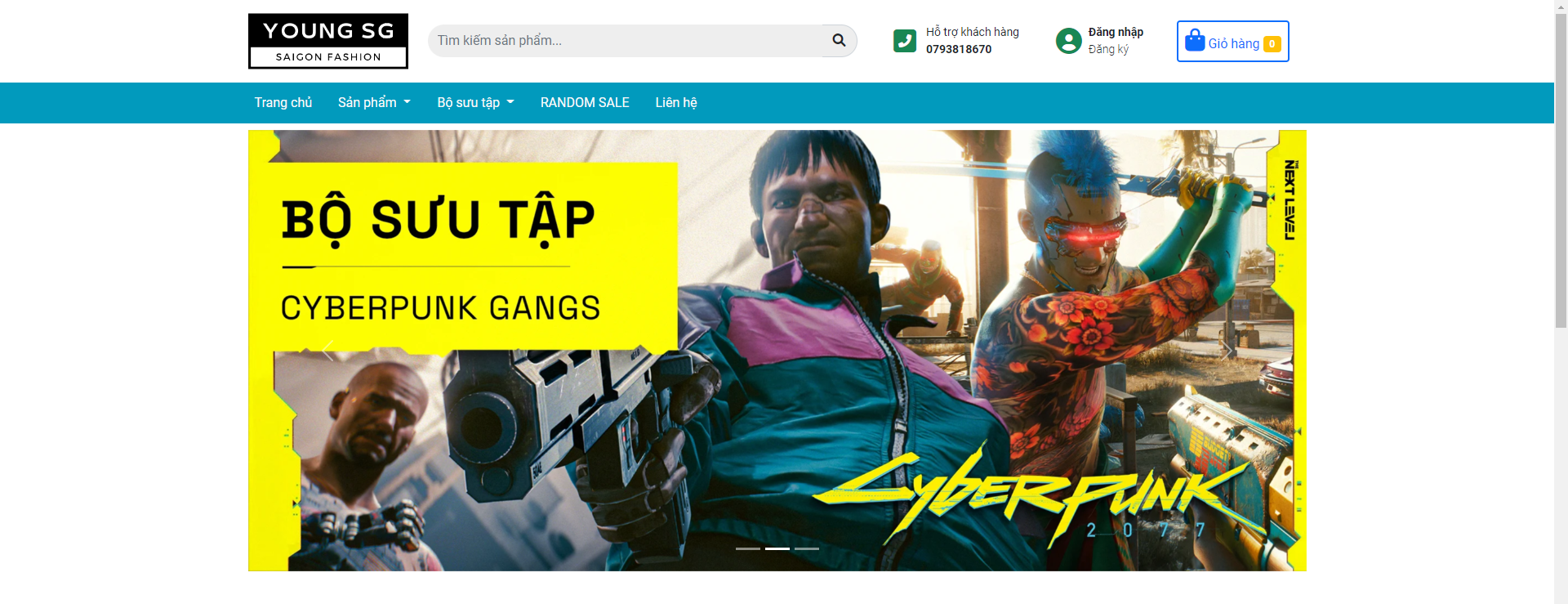


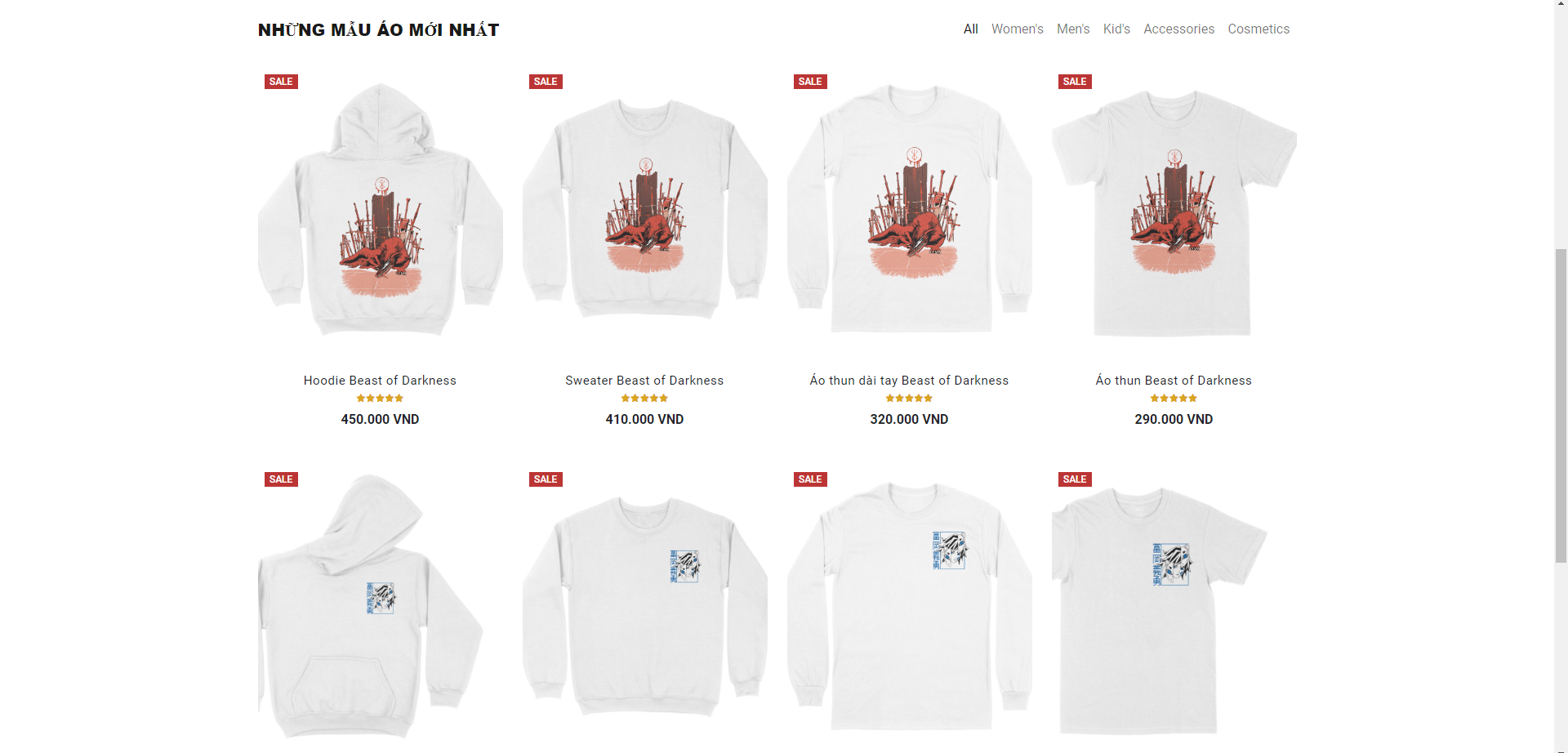
* 1. **Sơ đồ quan hệ thực thể:**

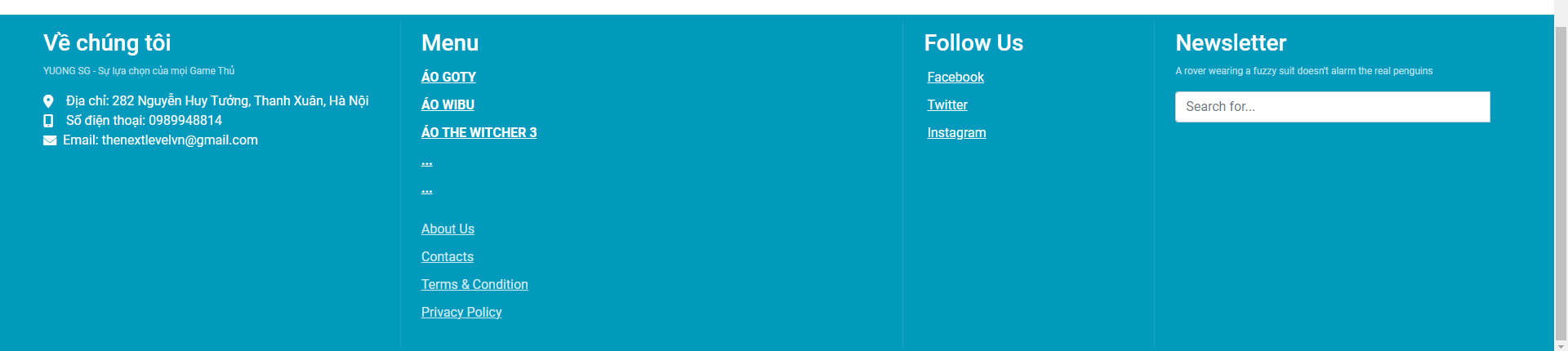
****

**Sơ đồ quan hệ thực thể bao gồm 9 bảng:**

1. Account
2. Products
3. Brands
4. Category
5. Color
6. Images
7. Size
8. Order
9. Order Details
10. **Thiết kế giao diện:**
11. **Trang chủ**

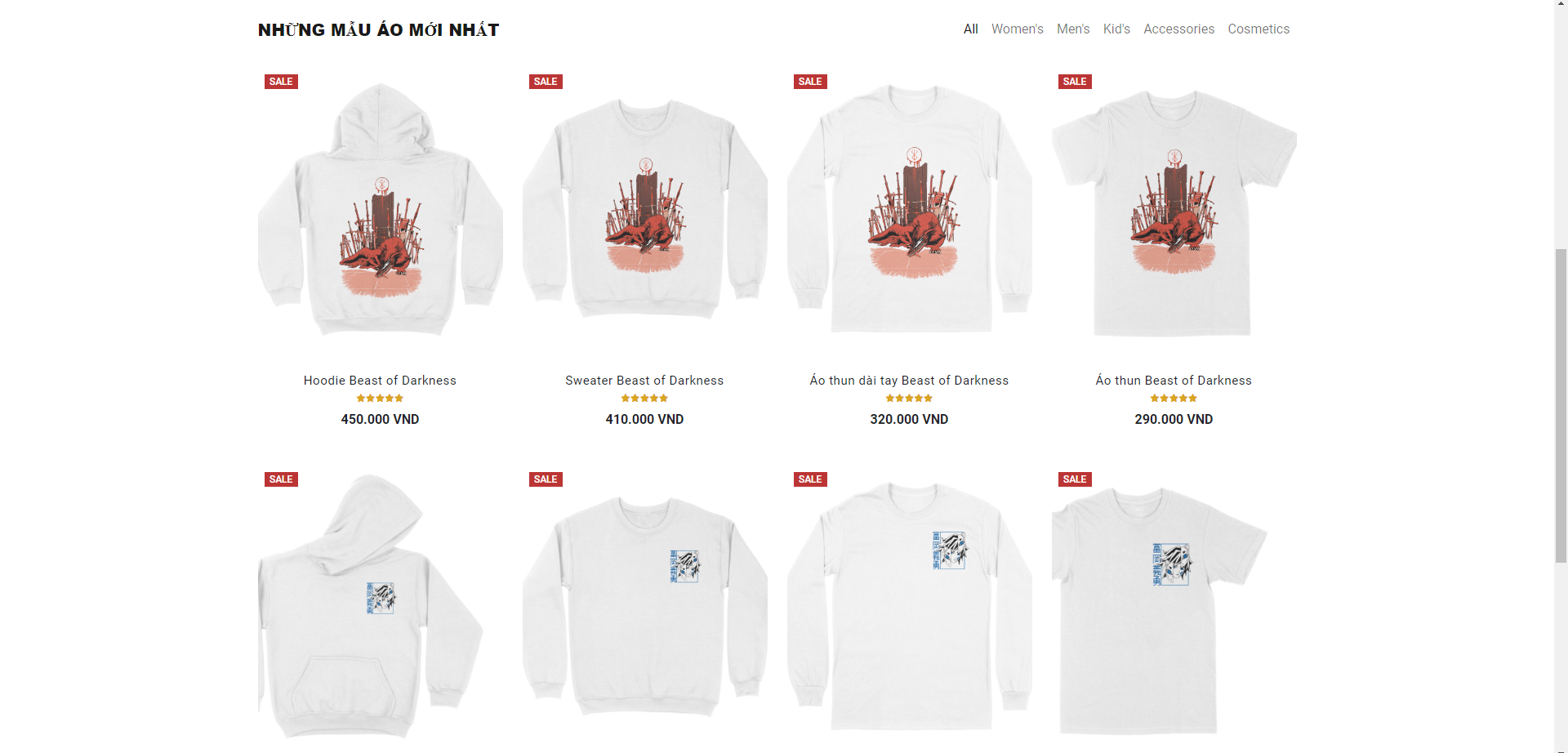






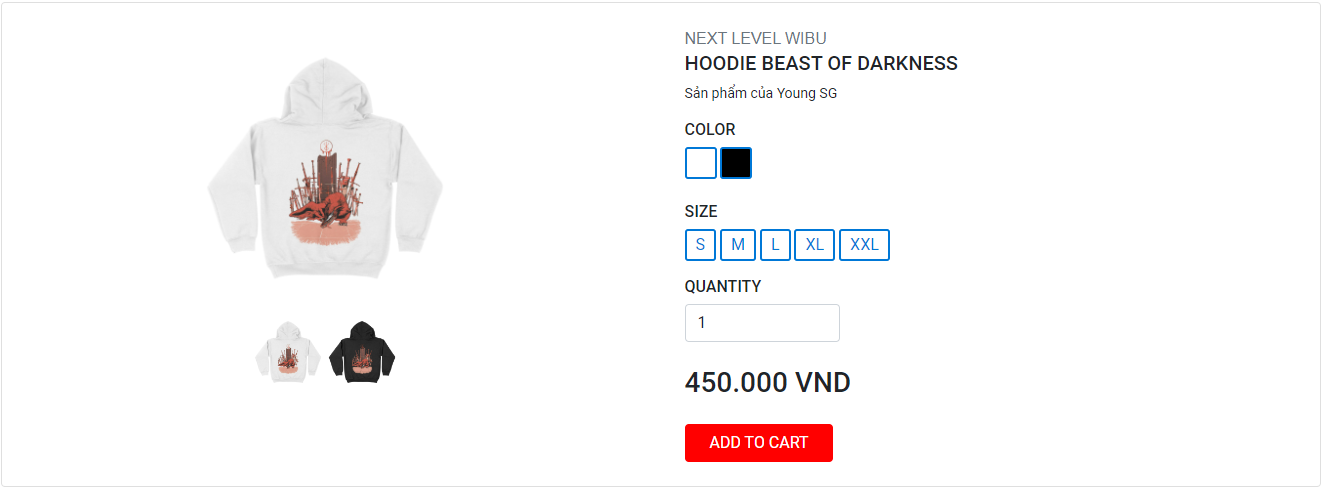
Hình ảnh trang chủ của website Young SG.

1. **Sản phẩm**



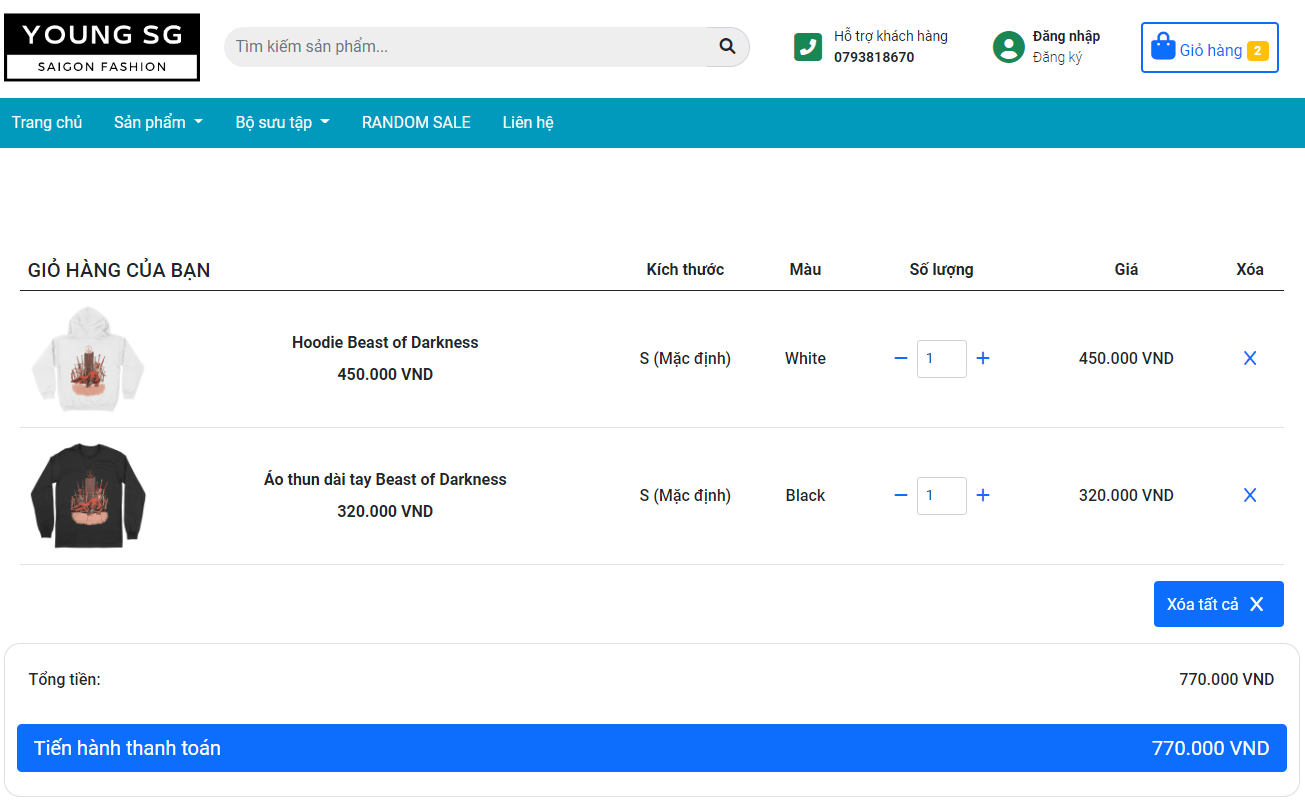
Sản phẩm được trình bày theo dạng thẻ.

1. **Chi tiết sản phẩm**



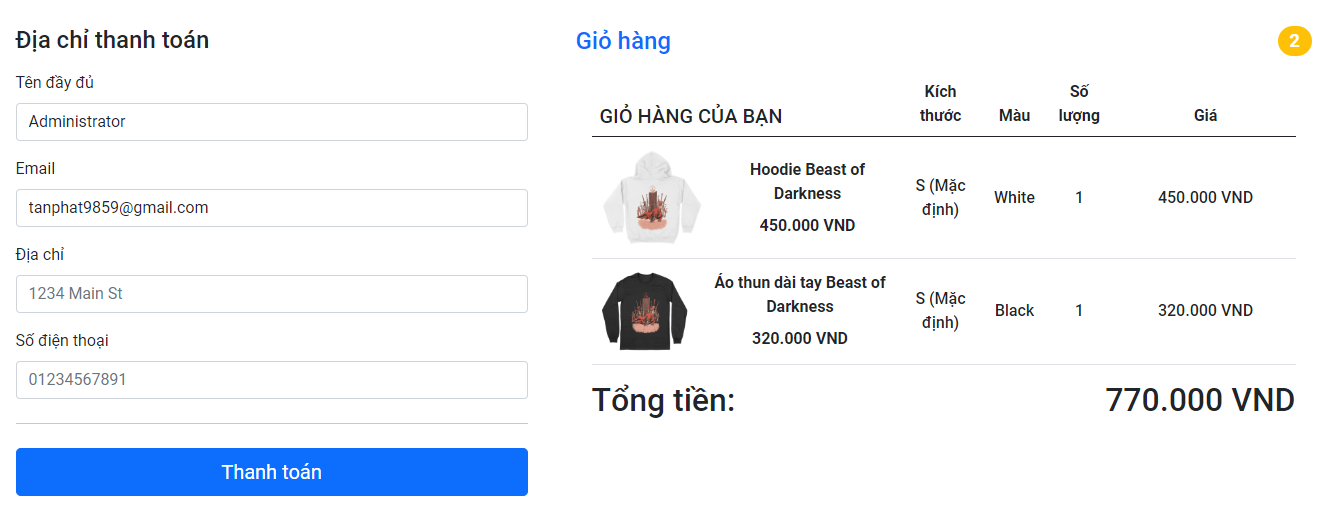
Chi tiết sản phẩm gồm có màu sắc, size áo, hình ảnh cụ thể.

1. **Giỏ hàng**

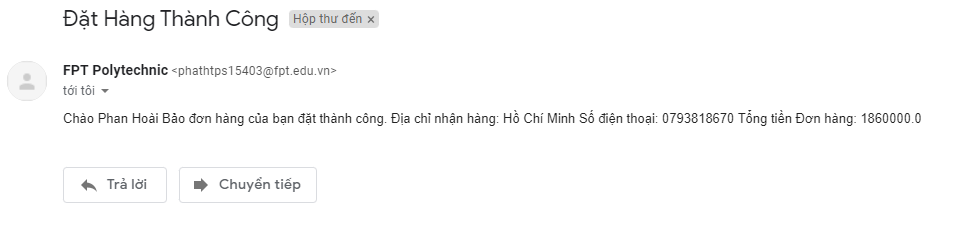


Người dùng chọn sản phẩm biểu tượng giỏ hàng sẽ cập nhật số lượng và bên trong sẽ có chi tiết sản phẩm đã chọn.

1. **Thanh toán**



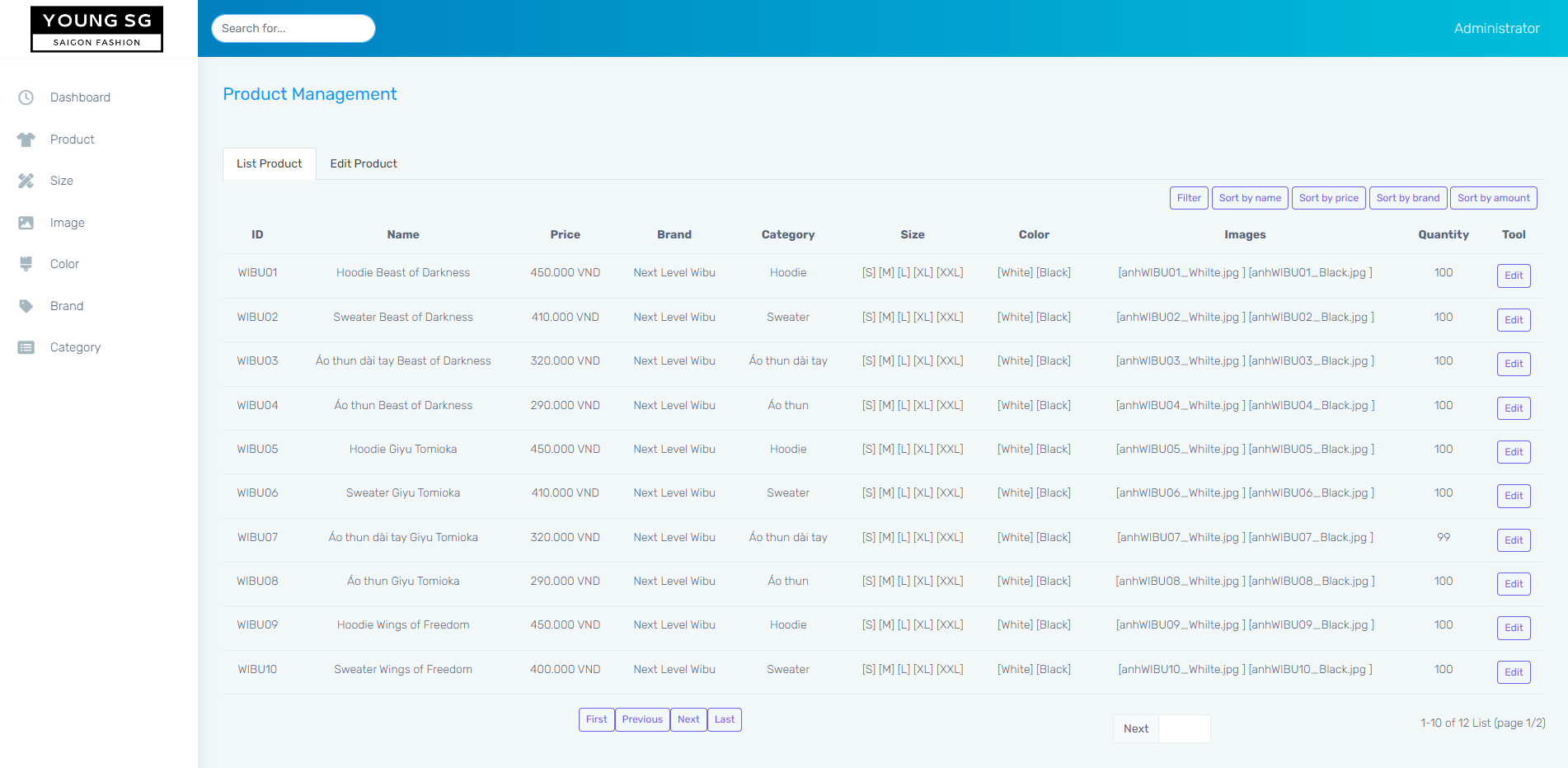
Tại form thanh toán người dùng sẽ điền vào những thông tin cần thiết để ghi nhận giao hàng, form sẽ hiển thị danh sách sản phẩm đã chọn.



Thanh toán sẽ gửi thông báo đến cho email của người dùng.

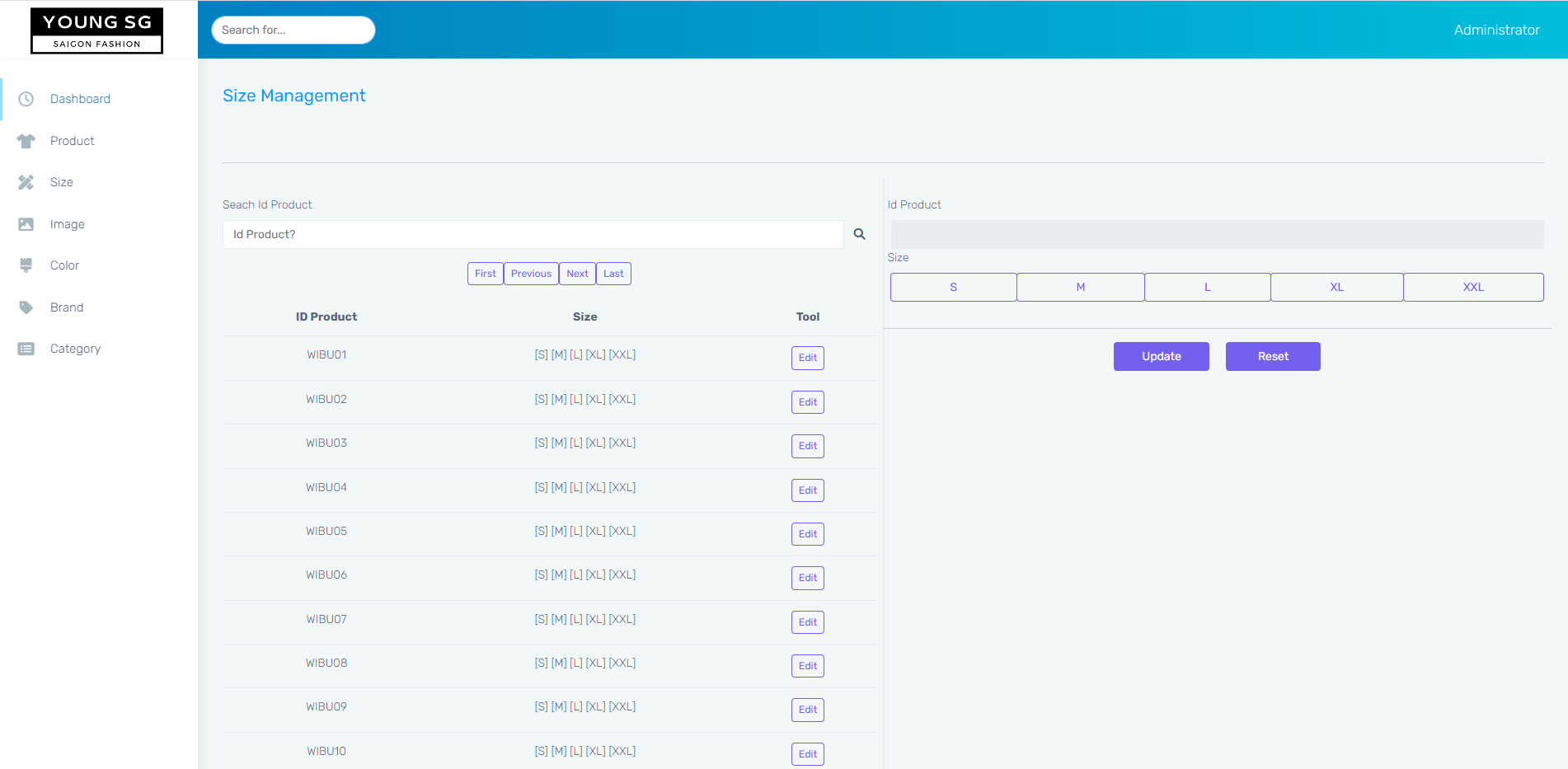
1. **Quản lý sản phẩm**

* Danh sách sản phẩm

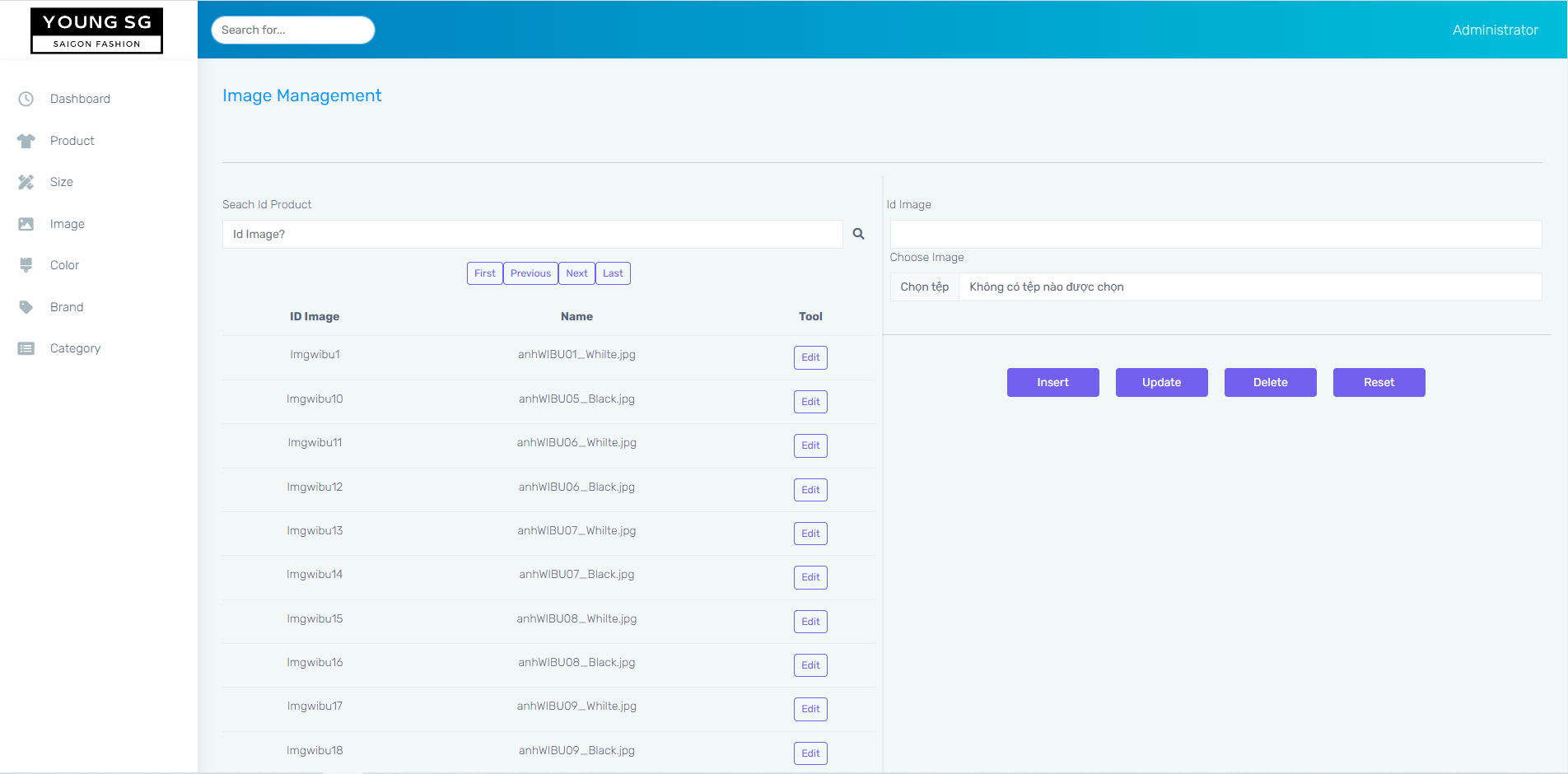


Với vai trò quản lý có thể xem được chi tiết danh sách sản phẩm đang có.

* Form chỉnh sửa thông tin sản phẩm



Chỉnh sửa size sản phẩm.



Chỉnh sửa hình ảnh của sản phẩm.

Chỉnh sửa màu sản phẩm.

Chỉnh sửa hãng.

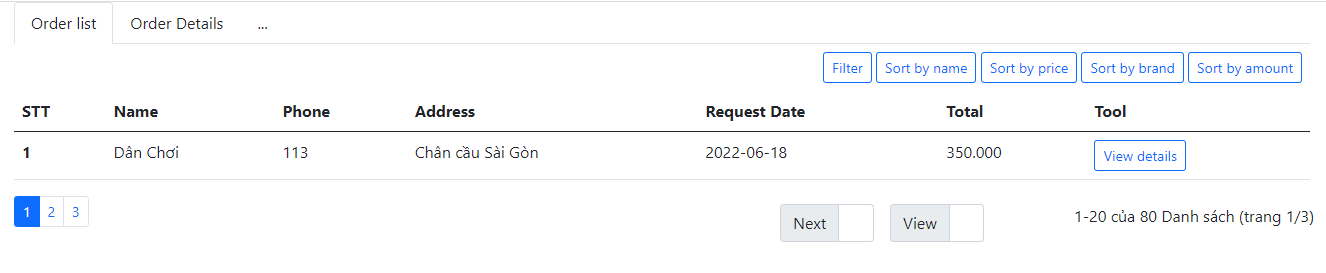
Chỉnh sửa số lượng sản phẩm.

1. **Quản lý người dùng**

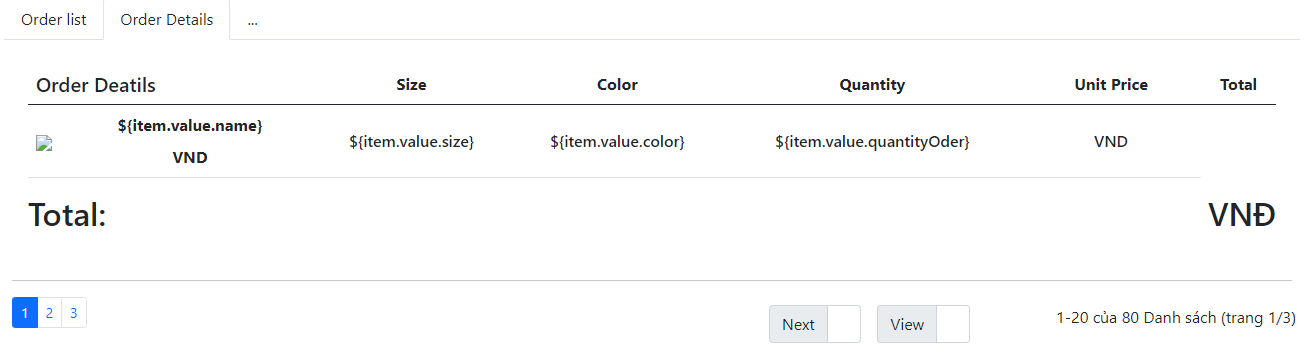
* Danh sách người dùng
* Form chỉnh sửa thông tin người dùng

1. **Thống kê**

* Danh sách hóa đơn



* Hóa đơn chi tiết



1. **THỰC THI VIẾT MÃ**
2. **Cơ sở dữ liệu**

* ***Khởi tạo các thực thể:***

1. USE [master]
2. GO
3. IF DB\_ID('ShopAoNhom1') **IS** NOT NULL **BEGIN**
4. **DROP** **DATABASE** ShopAoNhom1
5. **END**
6. **CREATE** **DATABASE** ShopAoNhom1
7. GO
8. USE [ShopAoNhom1]
9. GO
10. **CREATE** **TABLE** [dbo].[Products](
11. [Id] [**varchar**](10) NOT NULL,
12. [**Name**] [nvarchar](50) NOT NULL,
13. [Price] [**float**] NOT NULL,
14. [Info] [nvarchar](255),
15. [Available] [**bit**] NOT NULL,
16. [CategoryId] [**varchar**](10) NOT NULL,
17. [BrandId] [**varchar**](10) NOT NULL,
18. [Quantity] [**int**] NOT NULL,
19. **PRIMARY** **KEY** ([Id])
20. )
21. GO
22. **CREATE** **TABLE** [dbo].[Color](
23. [Id] [**int**] IDENTITY(1,1) NOT NULL,
24. [ProductId] [**varchar**](10) NOT NULL,
25. [ImageId] [**varchar**](10) NOT NULL,
26. [**Name**] [nvarchar](10) NOT NULL,
27. **PRIMARY** **KEY** ([Id],[ProductId]))
28. GO
29. **CREATE** **TABLE** [dbo].[Image](
30. [Id] [**varchar**](10) NOT NULL,
31. [**Name**] [nvarchar](50) NOT NULL,
32. **PRIMARY** **KEY** ([Id]))
33. GO
34. **CREATE** **TABLE** [dbo].[Brand](
35. [Id] [**varchar**](10) NOT NULL,
36. [**Name**] [nvarchar](50) NOT NULL,
37. **PRIMARY** **KEY** (Id))
38. GO
39. **CREATE** **TABLE** [dbo].[**Size**](
40. [Id] [**int**] IDENTITY(1,1) NOT NULL,
41. [ProductId] [**varchar**](10) NOT NULL,
42. [**Name**] [nvarchar](5) NOT NULL,
43. **PRIMARY** **KEY** ([Id],[ProductId]))
44. GO
45. **CREATE** **TABLE** [dbo].[Category](
46. [Id] [**varchar**](10) NOT NULL,
47. [**Name**] [nvarchar](50) NOT NULL,
48. **PRIMARY** **KEY** ([Id]))
49. GO
50. **CREATE** **TABLE** [dbo].[Accounts](
51. [Username] [nvarchar](50) NOT NULL,
52. [**Password**] [nvarchar](50) NOT NULL,
53. [Fullname] [nvarchar](50) NOT NULL,
54. [Email] [nvarchar](50) NOT NULL,
55. [Photo] [nvarchar](50) NOT NULL,
56. [Activated] [**bit**] NOT NULL,
57. [Admin] [**bit**] NOT NULL,
58. **PRIMARY** **KEY** ([Username]))
59. GO
61. **CREATE** **TABLE** [dbo].[OrderDetails](
62. [Id] [**bigint**] IDENTITY(1,1) NOT NULL,
63. [OrderId] [**bigint**] NOT NULL,
64. [ProductId] [**varchar**](10) NOT NULL,
65. [SizeId] [**int**] NOT NULL,
66. [ColorId] [**int**] NOT NULL,
67. [Price] [**float**] NOT NULL,
68. [Quantity] [**int**] NOT NULL,
69. **PRIMARY** **KEY** ([Id]))
70. GO
71. **CREATE** **TABLE** [dbo].[Orders](
72. [Id] [**bigint**] IDENTITY(1,1) NOT NULL,
73. [Username] [nvarchar](50) NOT NULL,
74. [CreateDate] [**date**] NOT NULL,
75. [Address] [nvarchar](100) NOT NULL,
76. [Phone] [nvarchar](11) NOT NULL,
77. [TotalMoney]  [**float**] NOT NULL,
78. **PRIMARY** **KEY** ([Id]))
80. **ALTER** **TABLE** [dbo].[Products]
81. **ADD** **CONSTRAINT** FK\_Product\_Brand
82. **FOREIGN** **KEY**([BrandId]) **REFERENCES** [dbo].[Brand]([Id]);
83. /\***ON** **UPDATE** **CASCADE** **ON** **DELETE** **CASCADE**;    \*/
84. GO
85. **ALTER** **TABLE** [dbo].[Products]
86. **ADD** **CONSTRAINT** FK\_Product\_Category
87. **FOREIGN** **KEY**([CategoryId]) **REFERENCES** [dbo].[Category]([Id]);
88. /\***ON** **UPDATE** **CASCADE** **ON** **DELETE** **CASCADE**;    \*/
89. GO
90. **ALTER** **TABLE** [dbo].[Color]
91. **ADD** **CONSTRAINT** FK\_Color\_Product
92. **FOREIGN** **KEY**([ProductId]) **REFERENCES** [dbo].[Products]([Id])
93. **ON** **UPDATE** **CASCADE** **ON** **DELETE** **CASCADE**;
94. GO
95. **ALTER** **TABLE** [dbo].[Color]
96. **ADD** **CONSTRAINT** FK\_Color\_Image
97. **FOREIGN** **KEY**([ImageId]) **REFERENCES** [dbo].[Image]([Id])
98. **ON** **UPDATE** **CASCADE** **ON** **DELETE** **CASCADE**;
99. GO
100. **ALTER** **TABLE** [dbo].[**Size**] **ADD** **CONSTRAINT** FK\_Size\_Product
101. **FOREIGN** **KEY**([ProductId]) **REFERENCES** [dbo].[Products]([Id])
102. **ON** **UPDATE** **CASCADE** **ON** **DELETE** **CASCADE**;
103. GO
105. **ALTER** **TABLE** [dbo].[OrderDetails]  **ADD** **CONSTRAINT** FK\_OrderDetails\_Orders
106. **FOREIGN** **KEY**([OrderId]) **REFERENCES** [dbo].[Orders] ([Id])
107. GO
108. **ALTER** **TABLE** [dbo].[OrderDetails]  **ADD** **CONSTRAINT** FK\_OrderDetails\_Product
109. **FOREIGN** **KEY**([ProductId]) **REFERENCES** [dbo].[Products] ([Id])
110. GO
111. **ALTER** **TABLE** [dbo].[OrderDetails]  **ADD** **CONSTRAINT** FK\_OrderDetails\_Size
112. **FOREIGN** **KEY**([SizeId],[ProductId]) **REFERENCES** [dbo].[**Size**] ([Id],[ProductId])
113. GO
114. **ALTER** **TABLE** [dbo].[OrderDetails]  **ADD** **CONSTRAINT** FK\_OrderDetails\_Color
115. **FOREIGN** **KEY**([ColorId],[ProductId]) **REFERENCES** [dbo].[Color] ([Id],[ProductId])
116. GO
117. **ALTER** **TABLE** [dbo].[Orders]  **ADD** **CONSTRAINT** FK\_Orders\_Accounts
118. **FOREIGN** **KEY**([Username]) **REFERENCES** [dbo].[Accounts] ([Username])

* ***Tạo dữ liệu mẫu:***

1. **INSERT** **INTO** [dbo].[Accounts]([Username],[**Password**],[Fullname],[Email],[Photo],[Activated],[Admin]) **VALUES**
2. ('user','123',N'Tấn Phát','phathtps15403@fpt.edu.vn','',1,0),
3. ('admin','123','Administrator','tanphat9859@gmail.com','',1,1)
4. GO
5. **INSERT** **INTO** [dbo].[Brand]([Id],[**Name**]) **VALUES**
6. ('NLW','Next Level Wibu')
7. GO
8. **INSERT** **INTO** [dbo].Category([Id],[**Name**]) **VALUES**
9. ('HD','Hoodie'),
10. ('SW','Sweater'),
11. ('ATDT',N'Áo thun dài tay'),
12. ('AT',N'Áo thun')
13. GO
14. **INSERT** **INTO** [dbo].[Products]([Id],[**Name**],[Price],[Info],[Available],[CategoryId],[BrandId],[Quantity]) **VALUES**
15. ('WIBU01',N'Hoodie Beast of Darkness',450000,N'Sản phẩm của Young SG','1','HD','NLW',100),
16. ('WIBU02',N'Sweater Beast of Darkness',410000,N'Sản phẩm của Young SG','1','SW','NLW',100),
17. ('WIBU03',N'Áo thun dài tay Beast of Darkness',320000,N'Sản phẩm của Young SG','1','ATDT','NLW',100),
18. ('WIBU04',N'Áo thun Beast of Darkness',290000,N'Sản phẩm của Young SG','1','AT','NLW',100),
19. ('WIBU05',N'Hoodie Giyu Tomioka',450000,N'Sản phẩm của Young SG','1','HD','NLW',100),
20. ('WIBU06',N'Sweater Giyu Tomioka',410000,N'Sản phẩm của Young SG','1','SW','NLW',100),
21. ('WIBU07',N'Áo thun dài tay Giyu Tomioka',320000,N'Sản phẩm của Young SG','1','ATDT','NLW',100),
22. ('WIBU08',N'Áo thun Giyu Tomioka',290000,N'Sản phẩm của Young SG','1','AT','NLW',100),
23. ('WIBU09',N'Hoodie Wings of Freedom',450000,N'Sản phẩm của Young SG','1','HD','NLW',100),
24. ('WIBU10',N'Sweater Wings of Freedom',400000,N'Sản phẩm của Young SG','1','SW','NLW',100),
25. ('WIBU11',N'Áo thun dài tay Wings of Freedom',320000,N'Sản phẩm của Young SG','1','ATDT','NLW',100),
26. ('WIBU12',N'Áo thun Wings of Freedom',290000,N'Sản phẩm của Young SG','1','AT','NLW',100)
28. GO
29. **INSERT** **INTO** [dbo].[**Size**]([ProductId],[**Name**]) **VALUES**
30. ('WIBU01','S'),
31. ('WIBU01','M'),
32. ('WIBU01','L'),
33. ('WIBU01','XL'),
34. ('WIBU01','XXL'),
35. ('WIBU02','S'),
36. ('WIBU02','M'),
37. ('WIBU02','L'),
38. ('WIBU02','XL'),
39. ('WIBU02','XXL'),
40. ('WIBU03','S'),
41. ('WIBU03','M'),
42. ('WIBU03','L'),
43. ('WIBU03','XL'),
44. ('WIBU03','XXL'),
45. ('WIBU04','S'),
46. ('WIBU04','M'),
47. ('WIBU04','L'),
48. ('WIBU04','XL'),
49. ('WIBU04','XXL'),
50. ('WIBU05','S'),
51. ('WIBU05','M'),
52. ('WIBU05','L'),
53. ('WIBU05','XL'),
54. ('WIBU05','XXL'),
55. ('WIBU06','S'),
56. ('WIBU06','M'),
57. ('WIBU06','L'),
58. ('WIBU06','XL'),
59. ('WIBU06','XXL'),
60. ('WIBU07','S'),
61. ('WIBU07','M'),
62. ('WIBU07','L'),
63. ('WIBU07','XL'),
64. ('WIBU07','XXL'),
65. ('WIBU08','S'),
66. ('WIBU08','M'),
67. ('WIBU08','L'),
68. ('WIBU08','XL'),
69. ('WIBU08','XXL'),
70. ('WIBU09','S'),
71. ('WIBU09','M'),
72. ('WIBU09','L'),
73. ('WIBU09','XL'),
74. ('WIBU09','XXL'),
75. ('WIBU10','S'),
76. ('WIBU10','M'),
77. ('WIBU10','L'),
78. ('WIBU10','XL'),
79. ('WIBU10','XXL'),
80. ('WIBU11','S'),
81. ('WIBU11','M'),
82. ('WIBU11','L'),
83. ('WIBU11','XL'),
84. ('WIBU11','XXL'),
85. ('WIBU12','S'),
86. ('WIBU12','M'),
87. ('WIBU12','L'),
88. ('WIBU12','XL'),
89. ('WIBU12','XXL')
90. GO
91. **INSERT** **INTO** [dbo].[Image]([Id],[**Name**]) **VALUES**
92. ('Imgwibu1','anhWIBU01\_Whilte.jpg'),
93. ('Imgwibu2','anhWIBU01\_Black.jpg'),
94. ('Imgwibu3','anhWIBU02\_Whilte.jpg'),
95. ('Imgwibu4','anhWIBU02\_Black.jpg'),
96. ('Imgwibu5','anhWIBU03\_Whilte.jpg'),
97. ('Imgwibu6','anhWIBU03\_Black.jpg'),
98. ('Imgwibu7','anhWIBU04\_Whilte.jpg'),
99. ('Imgwibu8','anhWIBU04\_Black.jpg'),
100. ('Imgwibu9','anhWIBU05\_Whilte.jpg'),
101. ('Imgwibu10','anhWIBU05\_Black.jpg'),
102. ('Imgwibu11','anhWIBU06\_Whilte.jpg'),
103. ('Imgwibu12','anhWIBU06\_Black.jpg'),
104. ('Imgwibu13','anhWIBU07\_Whilte.jpg'),
105. ('Imgwibu14','anhWIBU07\_Black.jpg'),
106. ('Imgwibu15','anhWIBU08\_Whilte.jpg'),
107. ('Imgwibu16','anhWIBU08\_Black.jpg'),
108. ('Imgwibu17','anhWIBU09\_Whilte.jpg'),
109. ('Imgwibu18','anhWIBU09\_Black.jpg'),
110. ('Imgwibu19','anhWIBU10\_Whilte.jpg'),
111. ('Imgwibu20','anhWIBU10\_Black.jpg'),
112. ('Imgwibu21','anhWIBU11\_Whilte.jpg'),
113. ('Imgwibu22','anhWIBU11\_Black.jpg'),
114. ('Imgwibu23','anhWIBU12\_Whilte.jpg'),
115. ('Imgwibu24','anhWIBU12\_Black.jpg')
116. GO
117. **INSERT** **INTO** [dbo].[Color]([ProductId],[ImageId],[**Name**]) **VALUES**
118. ('WIBU01','Imgwibu1','White'),
119. ('WIBU01','Imgwibu2','Black'),
120. ('WIBU02','Imgwibu3','White'),
121. ('WIBU02','Imgwibu4','Black'),
122. ('WIBU03','Imgwibu5','White'),
123. ('WIBU03','Imgwibu6','Black'),
124. ('WIBU04','Imgwibu7','White'),
125. ('WIBU04','Imgwibu8','Black'),
126. ('WIBU05','Imgwibu9','White'),
127. ('WIBU05','Imgwibu10','Black'),
128. ('WIBU06','Imgwibu11','White'),
129. ('WIBU06','Imgwibu12','Black'),
130. ('WIBU07','Imgwibu13','White'),
131. ('WIBU07','Imgwibu14','Black'),
132. ('WIBU08','Imgwibu15','White'),
133. ('WIBU08','Imgwibu16','Black'),
134. ('WIBU09','Imgwibu17','White'),
135. ('WIBU09','Imgwibu18','Black'),
136. ('WIBU10','Imgwibu19','White'),
137. ('WIBU10','Imgwibu20','Black'),
138. ('WIBU11','Imgwibu21','White'),
139. ('WIBU11','Imgwibu22','Black'),
140. ('WIBU12','Imgwibu23','White'),
141. ('WIBU12','Imgwibu24','Black')
142. **Front-end**

**HOME**

**Home (item)**

1. <%@ page language="java" contentType="text/html; charset=UTF-8"
2. pageEncoding="UTF-8"%>
3. <%@ taglib uri="http://java.sun.com/jstl/core\_rt" prefix="c"%>
4. <%@ taglib uri="http://java.sun.com/jsp/jstl/fmt" prefix="fmt"%>
5. <section>
6. <div **class**="container-xl m-auto">
7. <div id="carouselExampleIndicators" **class**="carousel slide"
8. data-bs-ride="carousel">
9. <div **class**="carousel-indicators">
10. <button type="button" data-bs-target="#carouselExampleIndicators"
11. data-bs-slide-to="0" **class**="active" aria-current="true"
12. aria-label="Slide 1"></button>
13. <button type="button" data-bs-target="#carouselExampleIndicators"
14. data-bs-slide-to="1" aria-label="Slide 2"></button>
15. <button type="button" data-bs-target="#carouselExampleIndicators"
16. data-bs-slide-to="2" aria-label="Slide 3"></button>
17. </div>
18. <div **class**="carousel-inner">
19. <div **class**="carousel-item active">
20. <img src="../images/p1.png" **class**="d-block w-100" alt="...">
21. </div>
22. <div **class**="carousel-item">
23. <img src="../images/p2.png" **class**="d-block w-100" alt="...">
24. </div>
25. <div **class**="carousel-item">
26. <img src="../images/p3.png" **class**="d-block w-100" alt="...">
27. </div>
28. </div>
29. <button **class**="carousel-control-prev" type="button"
30. data-bs-target="#carouselExampleIndicators" data-bs-slide="prev">
31. <span **class**="carousel-control-prev-icon" aria-hidden="true"></span>
32. <span **class**="visually-hidden">Previous</span>
33. </button>
34. <button **class**="carousel-control-next" type="button"
35. data-bs-target="#carouselExampleIndicators" data-bs-slide="next">
36. <span **class**="carousel-control-next-icon" aria-hidden="true"></span>
37. <span **class**="visually-hidden">Next</span>
38. </button>
39. </div>
40. </div>
41. </section>
42. <article **class**="container-xl">
43. <div>
44. <main>
45. <!-- DEMO HTML -->
46. <div **class**="container bg-white">
47. <nav **class**="navbar navbar-expand-md navbar-light bg-white">
48. <div **class**="container-fluid p-0">
49. <a **class**="navbar-brand text-uppercase fw-800" href="#"> <span
50. **class**="border-red pe-2 brand">NHỮNG MẪU ÁO MỚI NHẤT</span>
51. </a>
52. <button **class**="navbar-toggler" type="button"
53. data-bs-toggle="collapse" data-bs-target="#myNav"
54. aria-controls="myNav" aria-expanded="false"
55. aria-label="Toggle navigation">
56. <span **class**="fas fa-bars"></span>
57. </button>
58. <div **class**="collapse navbar-collapse" id="myNav">
59. <div **class**="navbar-nav ms-auto">
60. <a **class**="nav-link active" aria-current="page" href="#">All</a>
61. <a **class**="nav-link" href="#">Women's</a> <a **class**="nav-link"
62. href="#">Men's</a> <a class="nav-link" href="#">Kid's</a> <a
63. **class**="nav-link" href="#">Accessories</a> <a **class**="nav-link"
64. href="#">Cosmetics</a>
65. </div>
66. </div>
67. </div>
68. </nav>
69. <div **class**="row">
70. <c:forEach var="item" items="${page.content}">
71. <div
72. **class**="col-lg-3 col-sm-6 d-flex flex-column align-items-center justify-content-center product-item my-3">
73. <div **class**="product">
74. <img src="../images/${item.colors.get(0).getImage().getName() }"
75. alt="Hình ảnh không tồn tại">
76. <ul
77. **class**="d-flex align-items-center justify-content-center list-unstyled icons">
78. <li **class**="icon me-2"><a **class**="btn"
79. href="/product/detail?id=${item.id}"><i **class**="fas fa-eye"></i></a></li>
80. <li **class**="icon ms-2"><a **class**="btn"
81. data-bs-toggle="modal" data-bs-target="#${item.id}"> <span
82. **class**="fas fa-shopping-bag"></span>
83. </a></li>
84. </ul>
85. </div>
86. <div **class**="tag bg-red">sale</div>
87. <div **class**="title pt-4 pb-1">${item.name}</div>
88. <div **class**="d-flex align-content-center justify-content-center">
89. <span **class**="fas fa-star"></span> <span **class**="fas fa-star"></span>
90. <span **class**="fas fa-star"></span> <span **class**="fas fa-star"></span>
91. <span **class**="fas fa-star"></span>
92. </div>
93. <div **class**="price">
94. <fmt:formatNumber type="number" value="${item.price}" />
95. VND
96. </div>
97. </div>
98. <div **class**="modal fade" id="${item.id}" tabindex="-1"
99. aria-labelledby="exampleModalLabel" aria-hidden="true">
100. <div **class**="modal-dialog modal-dialog-centered modal-lg">
101. <div **class**="modal-content">
102. <div **class**="modal-body p-0 m-0">
103. <form **class**="" action="/cart/add/${item.id }/false"
104. method="get">
105. <div **class**="row d-flex justify-content-center">
106. <div **class**="col-md-12 p-0">
107. <div **class**="card">
108. <div **class**="row">
109. <div **class**="col-md-6 p-0">
110. <div **class**="images p-3">
111. <div **class**="text-center p-4">
112. <img id="main-image"
113. src="../images/${item.colors.get(0).getImage().getName() }"
114. width="250" />
115. </div>
116. <div **class**="thumbnail text-center">
117. <c:forEach var="image" items="${item.colors}">
118. <img onclick="change\_image(this)"
119. src="../images/${image.getImage().getName() }"
120. width="70">
122. </c:forEach>
123. </div>
124. </div>
125. </div>
126. <div **class**="col-md-6 p-0">
127. <div **class**="product p-4">
128. <div
129. **class**="d-flex justify-content-between align-items-center">
130. <button **class**="btn" data-bs-dismiss="modal">
131. <div **class**="d-flex align-items-center">
132. <i **class**="fa fa-long-arrow-left"></i>   <span
133. **class**="ml-1">Back</span>
134. </div>
135. </button>
136. </div>
137. <div **class**="mt-4 mb-3">
138. <span **class**="text-uppercase text-muted brand">${item.brand.getName()}</span>
139. <h5 **class**="text-uppercase">${item.name}</h5>
140. <div **class**="price d-flex flex-row align-items-center">
141. <span **class**="act-price"><fmt:formatNumber
142. type="number" value="${item.price}" /> VND</span>
143. <!-- <div **class**="ml-2">
144. <small **class**="dis-price">$59</small> <span>40%
145. OFF</span>
146. </div> -->
147. </div>
148. </div>
149. <p **class**="about">${item.info}</p>
150. <div **class**="mt-3">
151. <h6 **class**="text-uppercase">Color</h6>
152. <c:forEach var="color" items="${item.colors}">
153. <label **class**="rdocolor"> <input type="radio"
154. id="color" name="color" value="${color.name}">
155. <span **class**="bg-${color.name}"
156. style="height: 32px; width: 32px;"></span>
157. </label>
158. </c:forEach>
159. </div>
160. <div **class**="mt-3">
161. <h6 **class**="text-uppercase">Size</h6>
162. <c:forEach var="size" items="${item.sizes}">
163. <label **class**="radio"> <input type="radio"
164. id="size" name="size" value="${size.name}">
165. <span>${size.name}</span>
166. </label>
167. </c:forEach>
168. </div>
169. <div **class**="mt-3">
170. <h6 **class**="text-uppercase">Quantity</h6>
171. <input name="soLuongMua" type="number" value="1"
172. **class**="form-control w-25" placeholder="1" min="1"
173. max="${item.quantity}">
174. </div>
175. <div **class**="cart mt-4 align-items-center">
176. <button
177. **class**="btn btn-danger text-uppercase mr-2 px-4"
178. type="submit">Add to cart</button>
179. </div>
180. </div>
181. </div>
182. </div>
183. </div>
184. </div>
185. </div>
186. </form>
187. </div>
188. </div>
189. </div>
190. </div>
191. </c:forEach>
192. </div>
194. </div>
195. <!-- EMD DEMO HTML -->
196. </main>
197. </div>
198. </article>
199. <script>
200. function change\_image(image) {
201. var container = document.getElementById("main-image");
202. container.src = image.src;
203. }
204. </script>

**Home (index)**

1. <%@ page language="java" contentType="text/html; charset=UTF-8"
2. pageEncoding="UTF-8"%>
3. <%@ taglib uri="http://tiles.apache.org/tags-tiles" prefix="tiles"%>
4. <!DOCTYPE html>
5. <html dir="ltr" lang="en">
7. <head>
8. <meta charset="utf-8">
9. <meta http-equiv="X-UA-Compatible" content="IE=edge">
10. <!-- Tell the browser to be responsive to screen width -->
11. <meta name="viewport" content="width=device-width, initial-scale=1">
12. <meta name="keywords"
13. content="wrappixel, admin dashboard, html css dashboard, web dashboard, bootstrap 5 admin, bootstrap 5, css3 dashboard, bootstrap 5 dashboard, Monsterlite admin bootstrap 5 dashboard, frontend, responsive bootstrap 5 admin template, Monster admin lite design, Monster admin lite dashboard bootstrap 5 dashboard template">
14. <meta name="description"
15. content="Monster Lite is powerful and clean admin dashboard template, inpired from Bootstrap Framework">
16. <meta name="robots" content="noindex,nofollow">
17. <title>Admin</title>
18. <link
19. href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css"
20. rel="stylesheet"
21. integrity="sha384-EVSTQN3/azprG1Anm3QDgpJLIm9Nao0Yz1ztcQTwFspd3yD65VohhpuuCOmLASjC"
22. crossorigin="anonymous">
23. <link rel="stylesheet"
24. href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.0.0-beta2/css/all.min.css">
25. <link rel="canonical"
26. href="https://www.wrappixel.com/templates/monster-admin-lite/" />
27. <!-- Favicon icon -->
28. <!-- Custom CSS -->
29. <link href="../assets/plugins/chartist/dist/chartist.min.css"
30. rel="stylesheet">
31. <!-- Custom CSS -->
32. <link href="../css/style.min.css" rel="stylesheet">

35. </head>
37. <body>
38. <div id="main-wrapper" data-layout="vertical" data-navbarbg="skin6"
39. data-sidebartype="full" data-sidebar-position="absolute"
40. data-header-position="absolute" data-boxed-layout="full">
41. <!-- ============================================================== -->
42. <!-- Topbar header - style you can find in pages.scss -->
43. <!-- ============================================================== -->
44. <tiles:insertAttribute name="header" />
45. <!-- ============================================================== -->
46. <!-- End Topbar header -->
47. <!-- ============================================================== -->
48. <!-- ============================================================== -->
49. <!-- Left Sidebar - style you can find in sidebar.scss  -->
50. <!-- ============================================================== -->
51. <tiles:insertAttribute name="aside" />
52. <!-- ============================================================== -->
53. <!-- End Left Sidebar - style you can find in sidebar.scss  -->
54. <!-- ============================================================== -->
55. <div **class**="page-wrapper">
56. <!-- ============================================================== -->
57. <!-- Page wrapper  -->
58. <!-- ============================================================== -->
59. <tiles:insertAttribute name="navbar" />
60. <!-- ============================================================== -->
61. <!-- End Bread crumb and right sidebar toggle -->
62. <!-- ============================================================== -->
63. <!-- ============================================================== -->
64. <!-- Container fluid  -->
65. <!-- ============================================================== -->
66. <div **class**="page-breadcrumb" style="min-height: 800px;">
67. <tiles:insertAttribute name="body" />
68. </div>
70. <!-- ============================================================== -->
71. <!-- End Container fluid  -->
72. <!-- ============================================================== -->
73. <!-- ============================================================== -->
74. <!-- footer -->
75. <!-- ============================================================== -->
77. <tiles:insertAttribute name="footer" />
78. <!-- ============================================================== -->
79. <!-- End footer -->
80. <!-- ============================================================== -->
81. </div>
82. </div>
83. <!-- ============================================================== -->
84. <!-- End Page wrapper  -->
85. <!-- ============================================================== -->
87. <!-- ============================================================== -->
88. <!-- End Wrapper -->
89. <!-- ============================================================== -->
90. <!-- ============================================================== -->
91. <!-- All Jquery -->
92. <!-- ============================================================== -->
93. <script src="../assets/plugins/jquery/dist/jquery.min.js"></script>
94. <!-- Bootstrap tether Core JavaScript -->
95. <script
96. src="../assets/plugins/bootstrap/dist/js/bootstrap.bundle.min.js"></script>
97. <script src="js/app-style-switcher.js"></script>
98. <!--Wave Effects -->
99. <script src="js/waves.js"></script>
100. <!--Menu sidebar -->
101. <script src="js/sidebarmenu.js"></script>
102. <!--Custom JavaScript -->
103. <script src="js/custom.js"></script>
104. <!--This page JavaScript -->
105. <!--flot chart-->
106. <script src="../assets/plugins/flot/jquery.flot.js"></script>
107. <script
108. src="../assets/plugins/flot.tooltip/js/jquery.flot.tooltip.min.js"></script>
109. <script src="js/pages/dashboards/dashboard1.js"></script>
110. <script
111. src="https://cdn.jsdelivr.net/npm/@popperjs/core@2.9.2/dist/umd/popper.min.js"
112. integrity="sha384-IQsoLXl5PILFhosVNubq5LC7Qb9DXgDA9i+tQ8Zj3iwWAwPtgFTxbJ8NT4GN1R8p"
113. crossorigin="anonymous"></script>
114. <script
115. src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.min.js"
116. integrity="sha384-cVKIPhGWiC2Al4u+LWgxfKTRIcfu0JTxR+EQDz/bgldoEyl4H0zUF0QKbrJ0EcQF"
117. crossorigin="anonymous"></script>
118. </body>
120. </html>

**Menu**

1. <%@ page pageEncoding="utf-8"%>
2. <%@ taglib uri="http://www.springframework.org/tags" prefix="s"%>
3. <%@ taglib uri="http://java.sun.com/jstl/core\_rt" prefix="c"%>
4. <nav **class**="mt-2">
5. <div **class**="container-x d-flex flex-column">
6. <div **class**="navbar navbar-expand-lg p-0 d-flex justify-content-center">
7. <div **class**="row container-xxl p-0 ">
8. <!-- Button -->
9. <div **class**="col-2 col-lg-2 d-lg-none d-block">
10. <button **class**="navbar-toggler" type="button"
11. data-bs-toggle="offcanvas" data-bs-target="#offcanvasNavbar">
12. <span><i **class**="fas fa-bars"></i></span>
13. </button>
14. </div>
16. <!-- Logo -->
17. <div **class**="col col-lg-2 m-auto my-2 text-center">
18. <picture>
19. <source media="(max-width: 991px)" srcset="../images/logo.png">
20. <img **class**="img-fluid logo" src="../images/logo.png" alt="logo">
21. </picture>
22. </div>
24. <!-- Giỏ hàng -->
25. <div
26. **class**="col-2 col-lg-2 d-lg-none d-block p-0 d-flex justify-content-center">
27. <a type="button" **class**="btn btn-outline-primary border-2 p-2"
28. href="<c:url value='/cart/view' />"> <i
29. **class**="fas fa-shopping-bag fs-3"></i> <span
30. **class**=" d-none d-xl-inline">Giỏ hàng</span> <b>(<span
31. **class**="badge bg-warning m-auto" id="countTotal">
32. ${sessionScope.cart == **null** ? '0' : sessionScope.cart.tongsl}</span>)
33. </b>
34. </a>
35. </div>
37. <!-- Tìm tiếm -->
38. <div **class**="col-xl-5 col-lg-4 col-12 header-center"
39. id="search-header">
40. <form action="/search" method="get" **class**="input-group"
41. role="search">
42. <input type="text" name="q" value="" autocomplete="off"
43. placeholder="Tìm kiếm sản phẩm..."
44. **class**="border-0 rounded-pill rounded-end auto-search form-control "
45. required=""> <input type="hidden" name="type"
46. value="product"> <span
47. **class**="input-group-btn input-group-text p-0 rounded-pill rounded-start auto-search">
48. <button type="submit" **class**="btn h-100">
49. <span **class**="fa fa-search"></span>
50. </button>
51. </span>
52. </form>
53. </div>

56. <!-- Người dùng -->
57. <div **class**="col-xl-5 col-lg-4 col-12 m-0">
58. <ul **class**="d-flex my-auto p-0 justify-content-center">
60. <!-- Hỗ trợ khách hàng -->
61. <div **class**="d-flex m-auto">
62. <li **class**="d-flex align-self-center d-lg-flex d-none me-2">
63. <i **class**="fas fa-phone-square-alt fa-2x text-success"></i>
64. </li>
65. <li
66. **class**="m-1 d-flex flex-column align-self-center d-lg-flex d-none contact">
67. <span> Hỗ trợ khách hàng </span> <a href=""
68. **class**="text-decoration-none text-dark fw-bold">0793818670</a>
69. </li>
70. </div>
72. <!-- Tài khoản người dùng -->
73. <div **class**="d-flex m-auto">
74. <li **class**="d-flex align-self-center d-lg-flex d-none me-2">
75. <i **class**="fas fa-user-circle fa-2x text-success"></i>
76. </li>
77. <li
78. **class**=" d-flex flex-column align-self-center d-lg-flex d-none user"
79. ng-show="${ktdn }"><span
80. **class**="text-decoration-none text-dark fw-bold">Chào:
81. ${user.fullname }</span> <a
82. **class**="text-decoration-none text-dark fw-light register"
83. href="/account/logout">Đăng xuất</a></li>
84. <li
85. **class**=" d-flex flex-column align-self-center d-lg-flex d-none user"
86. ng-show="!${ktdn }"><a
87. **class**="text-decoration-none text-dark fw-bold"
88. href="/account/login">Đăng nhập</a> <a
89. **class**="text-decoration-none text-dark fw-light register"
90. href="/account/signup">Đăng ký</a></li>
91. </div>
93. <!-- Giỏ hàng -->
94. <div **class**="d-flex m-auto">
95. <li **class**="d-flex align-self-center d-none d-lg-inline"><a
96. type="button" **class**="btn btn-outline-primary border-2 p-2"
97. href="<c:url value='/cart/view' />"> <i
98. **class**="fas fa-shopping-bag fs-3"></i> <span
99. **class**=" d-none d-xl-inline">Giỏ hàng</span> <b><span
100. **class**="badge bg-warning m-auto" id="countTotal">
101. ${sessionScope.cart == **null** ? '0' : sessionScope.cart.tongsl}</span>
102. </b>
103. </a></li>
104. </div>
106. </ul>
107. </div>
108. </div>
109. </div>
111. <!-- Menu -->
112. <div **class**="my-2 ">
113. <nav
114. **class**="navbar menu navbar-expand-lg navbar-dark d-lg-flex d-none">
115. <div **class**="container-xl">
116. <div **class**="collapse navbar-collapse" id="navbarSupportedContent">
117. <ul **class**="navbar-nav me-auto mb-2 mb-lg-0">
118. <li **class**="nav-item me-3"><a
119. **class**="nav-link active text-light" aria-current="page"
120. href="/home/index">Trang chủ</a></li>
121. <li **class**="nav-item dropdownone me-3"><a
122. **class**="nav-link dropdown-toggle active text-light"
123. href="#sanpham" id="navbarDropdown"> Sản phẩm </a>
124. <ul **class**="dropdown-menu" aria-labelledby="navbarDropdown">
125. <c:forEach var="item" items="${categories}">
126. <li><a **class**="dropdown-item" href="#">${item.name }</a></li>
127. </c:forEach>
128. </ul></li>
129. <li **class**="nav-item dropdownone me-3"><a
130. **class**="nav-link dropdown-toggle active text-light" href="#"
131. id="navbarDropdown"> Bộ sưu tập </a>
132. <ul **class**="dropdown-menu" aria-labelledby="navbarDropdown">
133. <c:forEach var="item" items="${brands}">
134. <li><a **class**="dropdown-item" href="#">${item.name }</a></li>
135. </c:forEach>
137. </ul></li>
138. <li **class**="nav-item me-3"><a
139. **class**="nav-link active text-light" aria-current="page" href="#">RANDOM
140. SALE</a></li>
141. <li **class**="nav-item"><a **class**="nav-link active text-light"
142. aria-current="page" href="#">Liên hệ</a></li>
143. </div>
144. </div>
145. </nav>
146. </div>
147. </div>
148. </nav>

151. <div>
152. <div **class**="offcanvas offcanvas-start menu-canvas" tabindex="-1"
153. id="offcanvasNavbar" aria-labelledby="offcanvasNavbarLabel">
154. <div **class**="offcanvas-header bg-lightblue">
155. <h5 **class**="offcanvas-title" id="offcanvasNavbarLabel">
156. <ul **class**="d-flex flex-row m-0 p-0">
157. <li **class**="m-auto me-1 d-flex align-self-center"><i
158. **class**="fas fa-user-circle fa-3x text-light me-2"></i></li>
159. <li **class**="m-auto ms-1 d-flex flex-column align-self-center"
160. ng-show="${ktdn }"><span
161. **class**="text-decoration-none text-light fw-bold">Chào:
162. ${user.fullname }</span> <a
163. **class**="text-decoration-none text-light fw-light"
164. href="/account/logout">Đăng xuất</a></li>
165. <li **class**="m-auto ms-1 d-flex flex-column align-self-center"
166. ng-show="!${ktdn }"><a
167. **class**="text-decoration-none text-light fw-bold"
168. href="/account/login">Đăng nhập</a> <a
169. **class**="text-decoration-none text-light fw-light"
170. href="/account/signup">Đăng ký</a></li>
171. </ul>
172. </h5>
173. <button type="button" **class**="btn-close text-reset"
174. data-bs-dismiss="offcanvas" aria-label="Close"></button>
175. </div>
177. <!-- Sub menu offcanvas-->
178. <div **class**="offcanvas-body">
179. <nav **class**="navbar navbar-light text-center fs-5">
180. <ul **class**="navbar-nav flex-grow-1 pe-3 fw-bold">
181. <li **class**="nav-item navsubitem"><a **class**="nav-link text-dark"
182. aria-current="page" href="#!homepage">Trang chủ</a></li>
183. <!-- Dropdown helper start -->
184. <li **class**="nav-item dropdown navsubitem"><span
185. **class**="nav-link text-dark dropdown-toggle" href="#"
186. id="offcanvasNavbarDropdown" role="button"
187. data-bs-toggle="dropdown" aria-expanded="false"> Sản phẩm </span>
188. <ul **class**="dropdown-menu"
189. aria-labelledby="offcanvasNavbarDropdown">
190. <c:forEach var="item" items="${categories}">
191. <li><a **class**="dropdown-item" href="#">${item.name }</a></li>
192. </c:forEach>
193. </ul></li>
194. <!-- Dropdown helper end -->
195. <!-- Dropdown account start -->
196. <li **class**="nav-item dropdown navsubitem"><span
197. **class**="nav-link text-dark dropdown-toggle" href="#"
198. id="offcanvasNavbarDropdown" role="button"
199. data-bs-toggle="dropdown" aria-expanded="false"> Bộ sưu tập
200. </span>
201. <ul **class**="dropdown-menu"
202. aria-labelledby="offcanvasNavbarDropdown">
203. <c:forEach var="item" items="${brands}">
204. <li><a **class**="dropdown-item" href="#">${item.name }</a></li>
205. </c:forEach>
206. </ul></li>
207. <!-- Dropdown account end -->
208. <li **class**="nav-item navsubitem"><a **class**="nav-link text-dark"
209. aria-current="page" href="#!homepage">RANDOM SALE</a></li>
210. </ul>
211. </nav>
213. </div>
214. </div>
215. </div>

**Footer**

1. <%@ page language="java" contentType="text/html; charset=UTF-8"
2. pageEncoding="UTF-8"%>
3. <footer **class**="footer-bs m-0 py-2">
4. <div **class**="row">
5. <div **class**="col-md-3 footer-brand animated fadeInLeft">
6. <h3 **class**="title-menu clicked">
7. Về chúng tôi
8. </h3>
9. <p>
10. YUONG SG - Sự lựa chọn của mọi Game Thủ
11. </p>
12. <div **class**="d-flex flex-row">
13. <i **class**="fa fa-map-marker-alt mt-1"></i>
14. <div **class**="ms-3">Địa chỉ:
15. <span>282 Nguyễn Huy Tưởng, Thanh Xuân, Hà Nội</span>
16. </div>
17. </div>
18. <div **class**="d-flex flex-row">
19. <i **class**="fa fa-mobile-alt mt-1"></i>
20. <div **class**="ms-3">
21. Số điện thoại: <a **class**="link text-decoration-none link-light" title="0989948814"
22. href="tel:0989948814">0989948814</a>
23. </div>
24. </div>
25. <div **class**="d-flex flex-row">
26. <i **class**="fa fa-envelope mt-1"></i>
27. <div **class**="ms-2">
28. Email: <a title="thenextlevelvn@gmail.com" **class**="text-decoration-none link-light"
29. href="mailto:thenextlevelvn@gmail.com">thenextlevelvn@gmail.com</a>
30. </div>
31. </div>
32. </div>
33. <div **class**="col-md-4 footer-nav animated fadeInUp">
34. <h3>Menu</h3>
35. <div **class**="col-md-6">
36. <ul **class**="pages">
37. <li><a href="#">Áo GOTY</a></li>
38. <li><a href="#">Áo Wibu</a></li>
39. <li><a href="#">Áo The Witcher 3</a></li>
40. <li><a href="#">...</a></li>
41. <li><a href="#">...</a></li>
42. </ul>
43. </div>
44. <div **class**="col-md-6">
45. <ul **class**="list">
46. <li><a href="#">About Us</a></li>
47. <li><a href="#">Contacts</a></li>
48. <li><a href="#">Terms & Condition</a></li>
49. <li><a href="#">Privacy Policy</a></li>
50. </ul>
51. </div>
52. </div>
53. <div **class**="col-md-2 footer-social animated fadeInDown">
54. <h3>Follow Us</h3>
55. <ul>
56. <li><a href="#">Facebook</a></li>
57. <li><a href="#">Twitter</a></li>
58. <li><a href="#">Instagram</a></li>
59. </ul>
60. </div>
61. <div **class**="col-md-3 footer-ns animated fadeInRight">
62. <h3>Newsletter</h3>
63. <p>A rover wearing a fuzzy suit doesn't alarm the real penguins</p>
64. <p>
65. <div **class**="input-group">
66. <input type="text" **class**="form-control" placeholder="Search for...">
67. <span **class**="input-group-btn">
68. <button **class**="btn btn-default" type="button"><span **class**="glyphicon glyphicon-envelope"></span></button>
69. </span>
70. </div><!-- /input-group -->
71. </p>
72. </div>
73. </div>
74. </footer>

**ADMIN**

**Home (admin)**

1. <%@ page language="java" contentType="text/html; charset=UTF-8"
2. pageEncoding="UTF-8"%>
3. <%@ taglib uri="http://tiles.apache.org/tags-tiles" prefix="tiles"%>
4. <!DOCTYPE html>
5. <html dir="ltr" lang="en">
6. <head>
7. <meta charset="utf-8">
8. <meta http-equiv="X-UA-Compatible" content="IE=edge">
9. <!-- Tell the browser to be responsive to screen width -->
10. <meta name="viewport" content="width=device-width, initial-scale=1">
11. <meta name="keywords"
12. content="wrappixel, admin dashboard, html css dashboard, web dashboard, bootstrap 5 admin, bootstrap 5, css3 dashboard, bootstrap 5 dashboard, Monsterlite admin bootstrap 5 dashboard, frontend, responsive bootstrap 5 admin template, Monster admin lite design, Monster admin lite dashboard bootstrap 5 dashboard template">
13. <meta name="description"
14. content="Monster Lite is powerful and clean admin dashboard template, inpired from Bootstrap Framework">
15. <meta name="robots" content="noindex,nofollow">
16. <title>Admin</title>
17. <link
18. href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css"
19. rel="stylesheet"
20. integrity="sha384-EVSTQN3/azprG1Anm3QDgpJLIm9Nao0Yz1ztcQTwFspd3yD65VohhpuuCOmLASjC"
21. crossorigin="anonymous">
22. <link rel="stylesheet"
23. href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.0.0-beta2/css/all.min.css">
24. <link rel="canonical"
25. href="https://www.wrappixel.com/templates/monster-admin-lite/" />
26. <!-- Favicon icon -->
27. <!-- Custom CSS -->
28. <link href="../assets/plugins/chartist/dist/chartist.min.css"
29. rel="stylesheet">
30. <!-- Custom CSS -->
31. <link href="../css/style.min.css" rel="stylesheet">
32. </head>
33. <body>
34. <div id="main-wrapper" data-layout="vertical" data-navbarbg="skin6"
35. data-sidebartype="full" data-sidebar-position="absolute"
36. data-header-position="absolute" data-boxed-layout="full">
37. <tiles:insertAttribute name="header" />
38. <tiles:insertAttribute name="aside" />
39. <div **class**="page-wrapper">
40. <tiles:insertAttribute name="navbar" />
41. <div **class**="page-breadcrumb" style="min-height: 800px;">
42. <tiles:insertAttribute name="body" />
43. </div>
44. <tiles:insertAttribute name="footer" />
45. <script src="../assets/plugins/jquery/dist/jquery.min.js"></script>
46. <!-- Bootstrap tether Core JavaScript -->
47. <script
48. src="../assets/plugins/bootstrap/dist/js/bootstrap.bundle.min.js"></script>
49. <script src="js/app-style-switcher.js"></script>
50. <!--Wave Effects -->
51. <script src="js/waves.js"></script>
52. <!--Menu sidebar -->
53. <script src="js/sidebarmenu.js"></script>
54. <!--Custom JavaScript -->
55. <script src="js/custom.js"></script>
56. <!--This page JavaScript -->
57. <!--flot chart-->
58. <script src="../assets/plugins/flot/jquery.flot.js"></script>
59. <script
60. src="../assets/plugins/flot.tooltip/js/jquery.flot.tooltip.min.js"></script>
61. <script src="js/pages/dashboards/dashboard1.js"></script>
62. <script
63. src="https://cdn.jsdelivr.net/npm/@popperjs/core@2.9.2/dist/umd/popper.min.js"
64. integrity="sha384-IQsoLXl5PILFhosVNubq5LC7Qb9DXgDA9i+tQ8Zj3iwWAwPtgFTxbJ8NT4GN1R8p"
65. crossorigin="anonymous"></script>
66. <script
67. src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.min.js"
68. integrity="sha384-cVKIPhGWiC2Al4u+LWgxfKTRIcfu0JTxR+EQDz/bgldoEyl4H0zUF0QKbrJ0EcQF"
69. crossorigin="anonymous"></script>
70. </body>
71. </html>

**Dashboard (admin)**

1. <div **class**="container-fluid">
2. <!-- ============================================================== -->
3. <!-- Sales chart -->
4. <!-- ============================================================== -->
5. <div **class**="row">
6. <!-- Column -->
7. <div **class**="col-sm-6">
8. <div **class**="card">
9. <div **class**="card-body">
10. <h4 **class**="card-title">Daily Sales</h4>
11. <div **class**="text-end">
12. <h2 **class**="font-light mb-0">
13. <i **class**="ti-arrow-up text-success"></i> $120
14. </h2>
15. <span **class**="text-muted">Todays Income</span>
16. </div>
17. <span **class**="text-success">80%</span>
18. <div **class**="progress">
19. <div **class**="progress-bar bg-success" role="progressbar"
20. style="width: 80%; height: 6px;" aria-valuenow="25"
21. aria-valuemin="0" aria-valuemax="100"></div>
22. </div>
23. </div>
24. </div>
25. </div>
26. <!-- Column -->
27. <!-- Column -->
28. <div **class**="col-sm-6">
29. <div **class**="card">
30. <div **class**="card-body">
31. <h4 **class**="card-title">Weekly Sales</h4>
32. <div **class**="text-end">
33. <h2 **class**="font-light mb-0">
34. <i **class**="ti-arrow-up text-info"></i> $5,000
35. </h2>
36. <span **class**="text-muted">Todays Income</span>
37. </div>
38. <span **class**="text-info">30%</span>
39. <div **class**="progress">
40. <div **class**="progress-bar bg-info" role="progressbar"
41. style="width: 30%; height: 6px;" aria-valuenow="25"
42. aria-valuemin="0" aria-valuemax="100"></div>
43. </div>
44. </div>
45. </div>
46. </div>
47. <!-- Column -->
48. </div>
49. <!-- ============================================================== -->
50. <!-- Sales chart -->
51. <!-- ============================================================== -->
52. <div **class**="row">
53. <!-- column -->
54. <div **class**="col-sm-12">
55. <div **class**="card">
56. <div **class**="card-body">
57. <h4 **class**="card-title">Revenue Statistics</h4>
58. <div **class**="flot-chart">
59. <div **class**="flot-chart-content " id="flot-line-chart"
60. style="padding: 0px; position: relative;">
61. <canvas **class**="flot-base w-100" height="400"></canvas>
62. </div>
63. </div>
64. </div>
65. </div>
66. </div>
67. <!-- column -->
68. </div>
69. <!-- ============================================================== -->
70. <!-- Table -->
71. <!-- ============================================================== -->
72. <div **class**="row">
73. <div **class**="col-sm-12">
74. <div **class**="card">
75. <div **class**="card-body">
76. <div **class**="d-md-flex">
77. <h4 **class**="card-title col-md-10 mb-md-0 mb-3 align-self-center">Projects
78. of the Month</h4>
79. <div **class**="col-md-2 ms-auto">
80. <select **class**="form-select shadow-none col-md-2 ml-auto">
81. <option selected>January</option>
82. <option value="1">February</option>
83. <option value="2">March</option>
84. <option value="3">April</option>
85. </select>
86. </div>
87. </div>
88. <div **class**="table-responsive mt-5">
89. <table **class**="table stylish-table no-wrap">
90. <thead>
91. <tr>
92. <th **class**="border-top-0" colspan="2">Assigned</th>
93. <th **class**="border-top-0">Name</th>
94. <th **class**="border-top-0">Budget</th>
95. </tr>
96. </thead>
97. <tbody>
98. <tr>
99. <td style="width: 50px;"><span **class**="round">S</span></td>
100. <td **class**="align-middle">
101. <h6>Sunil Joshi</h6> <small **class**="text-muted">Web
102. Designer</small>
103. </td>
104. <td **class**="align-middle">Elite Admin</td>
105. <td **class**="align-middle">$3.9K</td>
106. </tr>
107. <tr **class**="active">
108. <td><span **class**="round"><img
109. src="../assets/images/users/2.jpg" alt="user" width="50"></span></td>
110. <td **class**="align-middle">
111. <h6>Andrew</h6> <small **class**="text-muted">Project
112. Manager</small>
113. </td>
114. <td **class**="align-middle">Real Homes</td>
115. <td **class**="align-middle">$23.9K</td>
116. </tr>
117. <tr>
118. <td><span **class**="round round-success">B</span></td>
119. <td **class**="align-middle">
120. <h6>Bhavesh patel</h6> <small **class**="text-muted">Developer</small>
121. </td>
122. <td **class**="align-middle">MedicalPro Theme</td>
123. <td **class**="align-middle">$12.9K</td>
124. </tr>
125. <tr>
126. <td><span **class**="round round-primary">N</span></td>
127. <td **class**="align-middle">
128. <h6>Nirav Joshi</h6> <small **class**="text-muted">Frontend
129. Eng</small>
130. </td>
131. <td **class**="align-middle">Elite Admin</td>
132. <td **class**="align-middle">$10.9K</td>
133. </tr>
134. <tr>
135. <td><span **class**="round round-warning">M</span></td>
136. <td **class**="align-middle">
137. <h6>Micheal Doe</h6> <small **class**="text-muted">Content
138. Writer</small>
139. </td>
140. <td **class**="align-middle">Helping Hands</td>
141. <td **class**="align-middle">$12.9K</td>
142. </tr>
143. <tr>
144. <td><span **class**="round round-danger">N</span></td>
145. <td **class**="align-middle">
146. <h6>Johnathan</h6> <small **class**="text-muted">Graphic</small>
147. </td>
148. <td **class**="align-middle">Digital Agency</td>
149. <td **class**="align-middle">$2.6K</td>
150. </tr>
151. </tbody>
152. </table>
153. </div>
154. </div>
155. </div>
156. </div>
157. </div>
158. <div **class**="row justify-content-center">
159. <!-- Column -->
160. <div **class**="col-lg-4 col-md-6">
161. <div **class**="card">
162. <img **class**="card-img-top img-responsive"
163. src="../assets/images/big/img1.jpg" alt="Card">
164. <div **class**="card-body">
165. <ul **class**="list-inline d-flex align-items-center">
166. <li **class**="ps-0">20 May 2021</li>
167. <li **class**="ms-auto"><a href="javascript:void(0)" **class**="link">3
168. Comment</a></li>
169. </ul>
170. <h3 **class**="font-normal">Featured Hydroflora Pots Garden &
171. Outdoors</h3>
172. </div>
173. </div>
174. </div>
175. <!-- Column -->
176. <!-- Column -->
177. <div **class**="col-lg-4 col-md-6">
178. <div **class**="card">
179. <img **class**="card-img-top img-responsive"
180. src="../assets/images/big/img2.jpg" alt="Card">
181. <div **class**="card-body">
182. <ul **class**="list-inline d-flex align-items-center">
183. <li **class**="ps-0">20 May 2021</li>
184. <li **class**="ms-auto"><a href="javascript:void(0)" **class**="link">3
185. Comment</a></li>
186. </ul>
187. <h3 **class**="font-normal">Featured Hydroflora Pots Garden &
188. Outdoors</h3>
189. </div>
190. </div>
191. </div>
192. <!-- Column -->
193. <!-- Column -->
194. <div **class**="col-lg-4 col-md-6">
195. <div **class**="card">
196. <img **class**="card-img-top img-responsive"
197. src="../assets/images/big/img4.jpg" alt="Card">
198. <div **class**="card-body">
199. <ul **class**="list-inline d-flex align-items-center">
200. <li **class**="ps-0">20 May 2021</li>
201. <li **class**="ms-auto"><a href="javascript:void(0)" **class**="link">3
202. Comment</a></li>
203. </ul>
204. <h3 **class**="font-normal">Featured Hydroflora Pots Garden &
205. Outdoors</h3>
206. </div>
207. </div>
208. </div>
209. <!-- Column -->
210. </div>
211. <!-- ============================================================== -->
212. <!-- Recent blogss -->
213. <!-- ============================================================== -->
214. </div>

**Navbar (admin)**

1. <div **class**="page-breadcrumb">
2. <div **class**="row align-items-center">
3. <div **class**="col-md-6 col-8 align-self-center">
4. <h3 **class**="page-title mb-0 p-0">${pageName }</h3>
5. <div **class**="d-flex align-items-center">
6. <%-- <nav aria-label="breadcrumb">
7. <ol **class**="breadcrumb">
8. <li **class**="breadcrumb-item"><a href="#">Home</a></li>
9. <li **class**="breadcrumb-item active" aria-current="page">${pageName }</li>
10. </ol>
11. </nav> --%>
12. </div>
13. </div>
14. </div>
15. </div>

**Header (admin)**

1. <%@ page language="java" contentType="text/html; charset=UTF-8"
2. pageEncoding="UTF-8"%>
3. <header **class**="topbar" data-navbarbg="skin6">
4. <nav **class**="navbar top-navbar navbar-expand-md navbar-dark">
5. <div **class**="navbar-header" data-logobg="skin6">
6. <!-- ============================================================== -->
7. <!-- Logo -->
8. <!-- ============================================================== -->
9. <a **class**="navbar-brand" href="/admin/index"> <!-- Logo icon --> <b
10. **class**="logo-icon text-center"> <!--You can put here icon as well // <i class="wi wi-sunset"></i> //-->
11. <!-- Dark Logo icon --> <img src="../images/logo.png"
12. alt="homepage" **class**="dark-logo w-75" />
13. </b>
14. </a>
15. <!-- ============================================================== -->
16. <!-- End Logo -->
17. <!-- ============================================================== -->
18. <!-- ============================================================== -->
19. <!-- toggle and nav items -->
20. <!-- ============================================================== -->
21. <a
22. **class**="nav-toggler waves-effect waves-light text-dark d-block d-md-none"
23. href="javascript:void(0)"><i **class**="ti-menu ti-close"></i></a>
24. </div>
25. <!-- ============================================================== -->
26. <!-- End Logo -->
27. <!-- ============================================================== -->
28. <div **class**="navbar-collapse collapse" id="navbarSupportedContent"
29. data-navbarbg="skin5">
31. <!-- ============================================================== -->
32. <!-- toggle and nav items -->
33. <!-- ============================================================== -->
34. <ul **class**="navbar-nav me-auto mt-md-0 ">
35. <!-- ============================================================== -->
36. <!-- Search -->
37. <!-- ============================================================== -->
39. <li **class**="nav-item hidden-sm-down">
40. <form **class**="app-search ps-3">
41. <input type="text" **class**="form-control"
42. placeholder="Search for..."> <a **class**="srh-btn"><i
43. **class**="ti-search"></i></a>
44. </form>
45. </li>
46. </ul>
47. <!-- ============================================================== -->
48. <!-- Right side toggle and nav items -->
49. <!-- ============================================================== -->

52. <ul **class**="navbar-nav">
53. <!-- ============================================================== -->
54. <!-- User profile and search -->
55. <!-- ============================================================== -->
56. <li **class**="nav-item dropdown">
57. <div **class**="collapse navbar-collapse" id="navbarNavDarkDropdown">
58. <ul **class**="navbar-nav">
59. <li **class**="nav-item dropdown"><a
60. **class**="nav-link dropdown-toggle" id="navbarDarkDropdownMenuLink"
61. role="button" data-bs-toggle="dropdown" aria-expanded="false">
62. ${user.fullname } </a>
63. <ul **class**="dropdown-menu dropdown-menu-dark"
64. aria-labelledby="navbarDarkDropdownMenuLink">
65. <li><a **class**="dropdown-item" href="/account/logout">Đăng
66. xuất</a></li>
67. </ul></li>
68. </ul>
69. </div>
70. </li>
71. </ul>
72. </div>
73. </nav>
74. </header>

**Aside (admin)**

1. <aside **class**="left-sidebar" data-sidebarbg="skin6">
2. <!-- Sidebar scroll-->
3. <div **class**="scroll-sidebar">
4. <!-- Sidebar navigation-->
5. <nav **class**="sidebar-nav">
6. <ul id="sidebarnav">
7. <!-- User Profile-->
8. <li **class**="sidebar-item"><a
9. **class**="sidebar-link waves-effect waves-dark sidebar-link"
10. href="/admin/index" aria-expanded="false"><i
11. **class**="me-3 far fa-clock fa-fw" aria-hidden="true"></i><span
12. **class**="hide-menu">Dashboard</span></a></li>
13. <li **class**="sidebar-item"><a
14. **class**="sidebar-link waves-effect waves-dark sidebar-link"
15. href="/admin/product-list" aria-expanded="false"> <i
16. **class**="me-3 fa-solid fa-shirt" aria-hidden="true"></i><span
17. **class**="hide-menu">Product</span></a></li>
19. <li **class**="sidebar-item"><a
20. **class**="sidebar-link waves-effect waves-dark sidebar-link"
21. href="/admin/size-editform" aria-expanded="false"><i
22. **class**="me-3 fa-solid fa-pen-ruler" aria-hidden="true"></i><span
23. **class**="hide-menu">Size</span></a></li>
25. <li **class**="sidebar-item"><a
26. **class**="sidebar-link waves-effect waves-dark sidebar-link"
27. href="/admin/image-editform" aria-expanded="false"><i
28. **class**="me-3 fa-solid fa-image" aria-hidden="true"></i><span
29. **class**="hide-menu">Image</span></a></li>
31. <li **class**="sidebar-item"><a
32. **class**="sidebar-link waves-effect waves-dark sidebar-link"
33. href="/admin/color-editform" aria-expanded="false"><i
34. **class**="me-3 fa-solid fa-brush" aria-hidden="true"></i><span
35. **class**="hide-menu">Color</span></a></li>
37. <li **class**="sidebar-item"><a
38. **class**="sidebar-link waves-effect waves-dark sidebar-link"
39. href="table-basic.html" aria-expanded="false"><i
40. **class**="me-3 fa-solid fa-tag" aria-hidden="true"></i><span
41. **class**="hide-menu">Brand</span></a></li>
42. <li **class**="sidebar-item"><a
43. **class**="sidebar-link waves-effect waves-dark sidebar-link"
44. href="icon-fontawesome.html" aria-expanded="false"><i
45. **class**="me-3 fa-solid fa-rectangle-list" aria-hidden="true"></i><span
46. **class**="hide-menu">Category</span></a></li>
47. </ul>
48. </nav>
49. <!-- End Sidebar navigation -->
50. </div>
51. <!-- End Sidebar scroll-->
52. </aside>

**Product list**

1. <%@ page language="java" contentType="text/html; charset=UTF-8"
2. pageEncoding="UTF-8"%>
3. <%@ taglib uri="http://java.sun.com/jstl/core\_rt" prefix="c"%>
4. <%@ taglib uri="http://java.sun.com/jsp/jstl/fmt" prefix="fmt"%>
5. <nav>
6. <div **class**="nav nav-tabs">
7. <a **class**="btn nav-link active text-dark" href="#">List Product</a> <a
8. **class**="btn nav-link text-dark" href="/admin/product-editform">Edit
9. Product</a>
10. </div>
11. </nav>
12. <div>
13. <div **class**="tab-pane fade show active" id="nav-list" role="tabpanel"
14. aria-labelledby="nav-home-tab">
15. <!-- Bộ công cụ -->
17. <div **class**="text-end p-2">
18. <a **class**="btn btn-outline-primary btn-sm"
19. href="/admin/product-list?field=id">Filter</a> <a
20. **class**="btn btn-outline-primary btn-sm"
21. href="/admin/product-list?field=name">Sort by name</a> <a
22. **class**="btn btn-outline-primary btn-sm"
23. href="/admin/product-list?field=price">Sort by price</a> <a
24. **class**="btn btn-outline-primary btn-sm"
25. href="/admin/product-list?field=brand">Sort by brand</a> <a
26. **class**="btn btn-outline-primary btn-sm"
27. href="/admin/product-list?field=quantity">Sort by amount</a>
28. </div>
29. <!-- Danh sách áo -->
30. <table **class**="table text-center">
31. <thead>
32. <tr>
33. <th scope="col">ID</th>
34. <th scope="col">Name</th>
35. <th scope="col">Price</th>
36. <th scope="col">Brand</th>
37. <th scope="col">Category</th>
38. <th scope="col">Size</th>
39. <th scope="col">Color</th>
40. <th scope="col">Images</th>
41. <th scope="col">Quantity</th>
42. <th scope="col">Tool</th>
43. </tr>
44. </thead>
45. <tbody>
46. <c:forEach var="item" items="${pageproduct.content}"
47. varStatus="loop">
48. <tr>
49. <td>${item.id }</td>
50. <td>${item.name }</td>
51. <td><fmt:formatNumber type="number" value="${item.price}" />
52. VND</td>
53. <td>${item.brand.name }</td>
54. <td>${item.category.name }</td>
55. <td><c:forEach var="size" items="${item.sizes }">
56. [<span>${size.name }</span>]</c:forEach></td>
57. <td><c:forEach var="color" items="${item.colors }">
58. [<span>${color.name }</span>]
59. </c:forEach></td>
60. <td><c:forEach var="color" items="${item.colors }">
61. [<span>${color.image.name } </span>]
62. </c:forEach></td>
63. <td>${item.quantity }</td>
64. <td><a **class**="btn btn-outline-primary btn-sm"
65. href="/admin/product-edit?id=${item.id }">Edit</a></td>
66. </tr>
67. </c:forEach>
68. </tbody>
69. </table>
70. <div **class**="row">
71. <div **class**="col">
72. <nav>
73. <ul **class**="pagination pagination-sm justify-content-center">
74. <li **class**="page-item"><a
75. **class**="btn btn-outline-primary btn-sm"
76. href="/admin/product-list?p=0">First</a></li>
77. <li **class**="page-item"><a
78. **class**="btn btn-outline-primary btn-sm"
79. href="/admin/product-list?p=${pageproduct.number-1}">Previous</a></li>
80. <li **class**="page-item"><a
81. **class**="btn btn-outline-primary btn-sm"
82. href="/admin/product-list?p=${pageproduct.number+1}">Next</a></li>
83. <li **class**="page-item"><a
84. **class**="btn btn-outline-primary btn-sm"
85. href="/admin/product-list?p=${pageproduct.totalPages-1}">Last</a></li>
86. </ul>
87. </nav>
88. </div>
90. <div **class**="col-1 text-end p-2">
91. <form action="/admin/product-list">
92. <div **class**="input-group mb-3">
93. <span **class**="input-group-text" id="basic-addon1">Next</span> <input
94. name="p" type="number" **class**="form-control" onchange="submit">
95. </div>
96. </form>
97. </div>
98. <div **class**="col-3 text-end p-2">
99. <span>1-10 of ${pageproduct.totalElements} List (page
100. ${pageproduct.number+1}/${pageproduct.totalPages})</span>
101. </div>
102. </div>
103. </div>
104. </div>

**Product editor**

1. <%@ page language="java" contentType="text/html; charset=UTF-8"
2. pageEncoding="UTF-8"%>
3. <%@ taglib uri="http://java.sun.com/jstl/core\_rt" prefix="c"%>
4. <%@ taglib uri="http://java.sun.com/jsp/jstl/fmt" prefix="fmt"%>
5. <%@ taglib uri="http://www.springframework.org/tags/form" prefix="form"%>
6. <nav>
7. <div **class**="nav nav-tabs">
8. <a **class**="btn nav-link text-dark" href="/admin/product-list">List
9. Product</a> <a **class**="btn nav-link active text-dark" href="#">Edit
10. Product</a>
11. </div>
12. </nav>
14. <div>
15. <form:form **class**="row g-3" action="/admin/product-editform"
16. modelAttribute="prod">
17. <form>
18. <div **class**="col-md-6">
19. <form:label **class**="form-label" path="id">ID </form:label>
20. <form:input **class**="form-control" path="id" id="id" />
21. </div>
22. <div **class**="col-md-6">
23. <form:label **class**="form-label" path="name">Name</form:label>
24. <form:input **class**="form-control" path="name" id="name" />
25. </div>
26. <div **class**="col-md-6">
27. <form:label **class**="form-label" path="price">Price</form:label>
28. <form:input **class**="form-control" path="price" id="price"
29. type="number" min="100000" max="5000000" />
30. </div>
31. <div **class**="col-md-6">
32. <form:label **class**="form-label" path="quantity">Quantity</form:label>
33. <form:input **class**="form-control" path="quantity" id="quantity"
34. type="number" min="1" max="500" />
35. </div>
37. <div **class**="col-md-6">
38. <form:label **class**="form-label" path="brand.id">Brand</form:label>
39. <form:select path="brand.id" **class**="form-select" id="brand.id">
40. <form:options items="${brands}" itemValue="id" itemLabel="name" />
41. </form:select>
42. </div>
43. <div **class**="col-md-6">
44. <form:label **class**="form-label" path="category.id">Category</form:label>
45. <form:select path="category.id" **class**="form-select" id="category.id">
46. <form:options items="${categorys}" itemValue="id" itemLabel="name" />
47. </form:select>
48. </div>
49. <div **class**="col-md-6 btn-group" role="group">
50. <form:label **class**="form-label m-auto me-2" path="available">Available</form:label>
52. <form:radiobutton path="available" value="true" **class**="btn-check"
53. name="btnradio" id="btnradio1" autocomplete="off" />
54. <form:label **class**="btn btn-outline-primary m-auto mx-auto"
55. **for**="btnradio1" path="available">Yes</form:label>
57. <form:radiobutton path="available" value="false" **class**="btn-check"
58. name="btnradio" id="btnradio2" autocomplete="off" />
59. <form:label **class**="btn btn-outline-primary m-auto mx-auto"
60. **for**="btnradio2" path="available">No</form:label>
62. </div>
64. <div **class**="col-md-6">
65. <form:label **class**="form-label" path="info">Info</form:label>
66. <form:textarea path="info" rows="3" cols="30" **class**="form-control"
67. id="info" />
68. </div>
70. <hr>
71. <!-- TOOL -->
72. <div **class**="row mt-4 justify-content-center">
73. <div **class**="col-1">
74. <button formaction="/admin/product-save"
75. **class**="btn btn-primary w-100">Insert</button>
76. </div>
77. <div **class**="col-1">
78. <button formaction="/admin/product-update"
79. **class**="btn btn-primary w-100">Update</button>
80. </div>
81. <div **class**="col-1">
82. <a href="/admin/product-delete?id=${prod.id }" type="button"
83. **class**="btn btn-primary w-100">Delete</a>
84. </div>
85. <div **class**="col-1">
86. <a href="/admin/product-editform" type="button"
87. **class**="btn btn-primary w-100">Reset</a>
88. </div>
89. </div>
90. </form>
91. </form:form>
92. </div>

**Image product editor**

1. <%@ page language="java" contentType="text/html; charset=UTF-8"
2. pageEncoding="UTF-8"%>
3. <%@ taglib uri="http://java.sun.com/jstl/core\_rt" prefix="c"%>
4. <%@ taglib uri="http://java.sun.com/jsp/jstl/fmt" prefix="fmt"%>
5. <%@ taglib uri="http://www.springframework.org/tags/form" prefix="form"%>
6. <hr>
7. <div **class**="row g-3">
8. <div **class**="col-md-6 border-end">
9. <div **class**="mt-4">
10. <label **class**="form-label me-2">Seach Id Product</label>
11. <form>
12. <div **class**="input-group mb-3">
13. <input name="idimgseach" type="text" value="" **class**="form-control"
14. placeholder="Id Image?" aria-label="Recipient's username"
15. aria-describedby="basic-addon2">
16. <button **class**="btn input-group-text" id="basic-addon2"
17. formaction="/admin/image-seachid">
18. <span><i **class**="fa-solid fa-magnifying-glass"></i></span>
19. </button>
21. </div>
22. </form>
23. </div>
24. <div **class**="col">
25. <nav>
26. <ul **class**="pagination pagination-sm justify-content-center">
27. <li **class**="page-item"><a
28. **class**="btn btn-outline-primary btn-sm"
29. href="/admin/image-editform?p=0">First</a></li>
30. <li **class**="page-item"><a
31. **class**="btn btn-outline-primary btn-sm"
32. href="/admin/image-editform?p=${page.number-1}">Previous</a></li>
33. <li **class**="page-item"><a
34. **class**="btn btn-outline-primary btn-sm"
35. href="/admin/image-editform?p=${page.number+1}">Next</a></li>
36. <li **class**="page-item"><a
37. **class**="btn btn-outline-primary btn-sm"
38. href="/admin/image-editform?p=${page.totalPages-1}">Last</a></li>
39. </ul>
40. </nav>
41. </div>
42. <table **class**="table text-center">
43. <thead>
44. <tr>
45. <th scope="col">ID Image</th>
46. <th scope="col">Name</th>
47. <th scope="col">Tool</th>
48. </tr>
49. </thead>
50. <tbody>
51. <c:forEach var="item" items="${page.content}">
52. <tr>
53. <td>${item.id }</td>
54. <td>${item.name }</td>
55. <td><a **class**="btn btn-outline-primary btn-sm"
56. href="/admin/size-edit?id=${item.id }">Edit</a></td>
57. </tr>
58. </c:forEach>
59. </tbody>
60. </table>
61. </div>
62. <div **class**="col-md-6 border-start">
63. <form>
64. <div **class**="row mt-4 justify-content-center">
65. <label **class**="form-label me-2">Id Image</label>
66. <div>
67. <input name="idimg" type="text"
68. value="${idimg }" **class**="form-control me-2"
69. id="exampleFormControlInput1">
70. </div>
71. <div **class**="mb-3">
72. <label **for**="formFile" **class**="form-label">Choose Image</label> <input
73. **class**="form-control" type="file" id="formFile">
74. </div>
75. </div>
77. <!-- TOOL -->
78. <div **class**="row mt-4 justify-content-center">
79. <hr>
80. <!-- TOOL -->
81. <div **class**="row mt-4 justify-content-center">
82. <div **class**="col-2">
83. <button formaction="/admin/product-save"
84. **class**="btn btn-primary w-100">Insert</button>
85. </div>
86. <div **class**="col-2">
87. <button formaction="/admin/product-update"
88. **class**="btn btn-primary w-100">Update</button>
89. </div>
90. <div **class**="col-2">
91. <a href="/admin/product-delete?id=${prod.id }" type="button"
92. **class**="btn btn-primary w-100">Delete</a>
93. </div>
94. <div **class**="col-2">
95. <a href="/admin/product-editform" type="button"
96. **class**="btn btn-primary w-100">Reset</a>
97. </div>
98. </div>
99. </div>
100. </form>
101. </div>

104. </div>

**Size product editor**

1. <%@ page language="java" contentType="text/html; charset=UTF-8"
2. pageEncoding="UTF-8"%>
3. <%@ taglib uri="http://java.sun.com/jstl/core\_rt" prefix="c"%>
4. <%@ taglib uri="http://java.sun.com/jsp/jstl/fmt" prefix="fmt"%>
5. <%@ taglib uri="http://www.springframework.org/tags/form" prefix="form"%>
6. <hr>
7. <div **class**="row g-3">
8. <div **class**="col-md-6 border-end">
9. <div **class**="mt-4">
10. <label **class**="form-label me-2">Seach Id Product</label>
11. <form>
12. <div **class**="input-group mb-3">
13. <input name="idproseach" type="text" value="" **class**="form-control"
14. placeholder="Id Product?" aria-label="Recipient's username"
15. aria-describedby="basic-addon2">
16. <button **class**="btn input-group-text" id="basic-addon2"
17. formaction="/admin/size-seachid">
18. <span><i **class**="fa-solid fa-magnifying-glass"></i></span>
19. </button>
21. </div>
22. </form>
23. </div>
24. <div **class**="col">
25. <nav>
26. <ul **class**="pagination pagination-sm justify-content-center">
27. <li **class**="page-item"><a
28. **class**="btn btn-outline-primary btn-sm"
29. href="/admin/size-editform?p=0">First</a></li>
30. <li **class**="page-item"><a
31. **class**="btn btn-outline-primary btn-sm"
32. href="/admin/size-editform?p=${pagesize.number-1}">Previous</a></li>
33. <li **class**="page-item"><a
34. **class**="btn btn-outline-primary btn-sm"
35. href="/admin/size-editform?p=${pagesize.number+1}">Next</a></li>
36. <li **class**="page-item"><a
37. **class**="btn btn-outline-primary btn-sm"
38. href="/admin/size-editform?p=${pagesize.totalPages-1}">Last</a></li>
39. </ul>
40. </nav>
41. </div>
42. <table **class**="table text-center">
43. <thead>
44. <tr>
45. <th scope="col">ID Product</th>
46. <th scope="col">Size</th>
47. <th scope="col">Tool</th>
48. </tr>
49. </thead>
50. <tbody>
51. <c:forEach var="item" items="${pagesize.content}">
52. <tr>
53. <td>${item.id }</td>
54. <td><c:forEach var="size" items="${item.sizes }">
55. [<span>${size.name }</span>]</c:forEach></td>
56. <td><a **class**="btn btn-outline-primary btn-sm"
57. href="/admin/size-edit?id=${item.id }">Edit</a></td>
58. </tr>
59. </c:forEach>
60. </tbody>
61. </table>
62. </div>
63. <div **class**="col-md-6 border-start">
64. <form>
65. <div **class**="row mt-4 justify-content-center">
66. <label **class**="form-label me-2">Id Product</label>
67. <div>
68. <input name="idprod" readonly="readonly" type="text"
69. value="${idprod }" **class**="form-control"
70. id="exampleFormControlInput1">
71. </div>
72. <label **class**="form-label me-2">Size</label>
73. <div **class**="btn-group" role="group"
74. aria-label="Basic checkbox toggle button group">
75. <input name="name" type="checkbox" **class**="btn-check" id="S"
76. autocomplete="off" value="S"> <label
77. **class**="btn btn-outline-primary" **for**="S">S</label> <input
78. name="name" type="checkbox" **class**="btn-check" id="M"
79. autocomplete="off" value="M"> <label
80. **class**="btn btn-outline-primary" **for**="M">M</label> <input
81. name="name" type="checkbox" **class**="btn-check" id="L"
82. autocomplete="off" value="L"> <label
83. **class**="btn btn-outline-primary" **for**="L">L</label> <input
84. name="name" type="checkbox" **class**="btn-check" id="XL"
85. autocomplete="off" value="XL"> <label
86. **class**="btn btn-outline-primary" **for**="XL">XL</label> <input
87. name="name" type="checkbox" **class**="btn-check" id="XXL"
88. autocomplete="off" value="XXL"> <label
89. **class**="btn btn-outline-primary" **for**="XXL">XXL</label>
90. </div>
91. </div>
93. <!-- TOOL -->
94. <div **class**="row mt-4 justify-content-center">
95. <hr>
96. <div **class**="col-2">
97. <button formaction="/admin/size-save" **class**="btn btn-primary w-100">Update</button>
98. </div>
99. <div **class**="col-2">
100. <a href="/admin/size-editform" type="button"
101. **class**="btn btn-primary w-100">Reset</a>
102. </div>
103. </div>
104. </form>
105. </div>
106. </div>

**PRODUCT**

**Details**

1. <%@ page language="java" contentType="text/html; charset=UTF-8"
2. pageEncoding="UTF-8"%>
3. <%@ taglib uri="http://java.sun.com/jstl/core\_rt" prefix="c"%>
4. <%@ taglib uri="http://java.sun.com/jsp/jstl/fmt" prefix="fmt"%>
5. <div **class**="container-xl m-auto">
6. <form **class**="" action="/cart/add/${item.id }/false" method="get">
7. <div **class**="row d-flex justify-content-center">
8. <div **class**="col-md-12 p-0">
9. <div **class**="card">
10. <div **class**="row">
11. <div **class**="col-md-6 p-0">
12. <div **class**="images p-3">
13. <div **class**="text-center p-4">
14. <img id="main-image"
15. src="../images/${item.colors.get(0).getImage().getName() }"
16. width="250" />
17. </div>
19. <div **class**="thumbnail text-center">
20. <c:forEach var="image" items="${item.colors}">
21. <img onclick="change\_image(this)"
22. src="../images/${image.getImage().getName() }" width="70">
23. </c:forEach>
24. </div>
26. </div>
27. </div>
28. <div **class**="col-md-6 p-0">
29. <div **class**="product p-4">
30. <div **class**="">
31. <span **class**="text-uppercase text-muted brand">${item.brand.getName()}</span>
32. <h5 **class**="text-uppercase">${item.name}</h5>
33. </div>
34. <p **class**="about">${item.info}</p>
35. <div **class**="mt-3">
36. <h6 **class**="text-uppercase">Color</h6>
37. <c:forEach var="color" items="${item.colors}">
38. <label **class**="rdocolor"> <input type="radio"
39. id="color" name="color" value="${color.name}"> <span
40. **class**="bg-${color.name}" style="height: 32px; width: 32px;"></span>
41. </label>
42. </c:forEach>
43. </div>
44. <div **class**="mt-3">
45. <h6 **class**="text-uppercase">Size</h6>
46. <c:forEach var="size" items="${item.sizes}">
47. <label **class**="radio"> <input type="radio" id="size"
48. name="size" value="${size.name}"> <span>${size.name}</span>
49. </label>
50. </c:forEach>
51. </div>
52. <div **class**="mt-3">
53. <h6 **class**="text-uppercase">Quantity</h6>
54. <input name="soLuongMua" type="number" value="1" **class**="form-control w-25"
55. placeholder="1" min="1" max="${item.quantity}">
56. </div>
57. <div **class**="price mt-4 d-flex flex-row align-items-center">
58. <h3 **class**="act-price m-0">
59. <fmt:formatNumber type="number" value="${item.price}" />
60. VND
61. </h3>
62. <!-- <div **class**="ml-2">
63. <small **class**="dis-price">$59</small> <span>40%
64. OFF</span>
65. </div> -->
66. </div>
67. <div **class**="cart mt-4 align-items-center">
68. <button **class**="btn btn-danger text-uppercase mr-2 px-4"
69. type="submit">Add to cart</button>
70. </div>
71. </div>
72. </div>
73. </div>
74. </div>
75. </div>
76. </div>
77. </form>
78. </div>

**Shoppingcart**

1. <%@ page language="java" contentType="text/html; charset=UTF-8"
2. pageEncoding="UTF-8"%>
3. <%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c"%>
4. <%@ taglib uri="http://java.sun.com/jsp/jstl/fmt" prefix="fmt"%>
6. <section **class**="h-100 h-custom">
7. <div **class**="container h-100 py-5">
8. <div
9. **class**="row d-flex justify-content-center align-items-center h-100">
10. <div **class**="col" id="listCart">
11. <div **class**="table-responsive m-3">
12. <table **class**="table text-center">
13. <thead>
14. <tr>
15. <th scope="col" **class**="text-start h5">GIỎ HÀNG CỦA BẠN</th>
16. <th scope="col">Kích thước</th>
17. <th scope="col">Màu</th>
18. <th scope="col">Số lượng</th>
19. <th scope="col">Giá</th>
20. <th scope="col">Xóa</th>
21. </tr>
22. </thead>
23. <tbody>
24. <!-- Start item -->
25. <c:forEach var="item" items="${sessionScope.cart.detail}"
26. varStatus="loop">
27. <tr id="tr-${item.value.id}">
28. <th scope="row">
29. <div **class**="d-flex align-items-center">
30. <div>
31. <img src="../images/${item.value.image}"
32. **class**="img-fluid rounded-3" style="width: 120px" />
33. </div>
34. <div **class**="flex-fill">
35. <div **class**="flex-column ms-4">
36. <p **class**="mb-2">${item.value.name}</p>
37. <p **class**="mb-0">
38. <fmt:formatNumber type="number"
39. value="${item.value.price}" />
40. VND
41. </p>
42. </div>
43. </div>
45. </div>
46. </th>
47. <td **class**="align-middle">
48. <p **class**="mb-0" style="font-weight: 500">${item.value.size}</p>
49. </td>
50. <td **class**="align-middle">
51. <p **class**="mb-0" style="font-weight: 500">${item.value.color}</p>
52. </td>
53. <td **class**="align-middle">
54. <form action="/cart/update/${item.value.id}/true" method="get">
55. <div **class**="d-flex flex-row justify-content-center">
56. <button **class**="btn btn-link px-2"
57. onmousedown="this.parentNode.querySelector('input[type=number]').stepDown()"
58. onmouseup="summit">
59. <i **class**="fas fa-minus"></i>
60. </button>
61. <input name="soLuongMua" onchange="summit" type="number"
62. **class**="form-control form-control-sm quantityProd"
63. value="${item.value.quantityOder}" style="width: 50px">
64. <button **class**="btn btn-link px-2"
65. onmousedown="this.parentNode.querySelector('input[type=number]').stepUp()"
66. onmouseup="summit">
67. <i **class**="fas fa-plus"></i>
68. </button>
69. </div>
70. </form>
72. </td>
73. <td **class**="align-middle" style="width: 180px">
74. <p **class**="mb-0" style="font-weight: 500"
75. id="totalSP-${cartDetail.value.masp}">
76. <fmt:formatNumber type="number"
77. value="${item.value.price \* item.value.quantityOder}" />
78. VND
79. </p>
80. </td>
81. <td **class**="align-middle"><a **class**="btn btn-link px-2"
82. href="/cart/remove/${item.value.id}"> <i
83. **class**="fa-solid fa-x"></i>
84. </a></td>
85. </tr>
86. <input type="hidden" id="giaSP-${item.value.id}"
87. value="${item.value.price}" />
88. <input type="hidden" id="colorSP-${item.value.id}"
89. value="${item.value.color}" />
90. <input type="hidden" id="sizeSP-${item.value.id}"
91. value="${item.value.size}" />
92. </c:forEach>
93. <!-- End item -->
94. </tbody>
95. </table>
96. <div **class**="text-end">
97. <a **class**="btn bg-primary text-white" href="/cart/clear"> Xóa
98. tất cả <i **class**="p-2 fa-solid fa-x"></i>
99. </a>
100. </div>
101. </div>
102. <!-- Start pay -->
103. <div
104. **class**="card pay-card shadow-2-strong mb-5 mb-lg-0 col-auto m-auto"
105. style="border-radius: 16px">
106. <div **class**="card-body p-4">
107. <div **class**="row">
108. <div **class**="d-flex justify-content-between mb-4"
109. style="font-weight: 500">
110. <p **class**="mb-2">Tổng tiền:</p>
111. <p **class**="mb-2" id="tongTien">
112. <fmt:formatNumber type="number"
113. value="${sessionScope.cart.tongtien}" />
114. VND
115. </p>
116. </div>
118. <a type="button" **class**="btn btn-primary btn-block btn-lg"
119. id="payNow" href="/cart/checkout"> <span
120. **class**="d-flex justify-content-between"> <span>Tiến
121. hành thanh toán</span> <span id="tongTien"><fmt:formatNumber
122. type="number" value="${sessionScope.cart.tongtien}" /> VND</span>
123. </span>
124. </a>
125. </div>
126. </div>
127. </div>
128. </div>
129. <!-- End pay -->
130. </div>
131. </div>
132. </section>
133. <script
134. src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.0/jquery.min.js"></script>

**Checkout**

1. <%@ page language="java" contentType="text/html; charset=UTF-8"
2. pageEncoding="UTF-8"%>
3. <%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c"%>
4. <%@ taglib uri="http://java.sun.com/jsp/jstl/fmt" prefix="fmt"%>
5. <div **class**="container">
6. <div **class**="row g-5">
7. <div **class**="col-md-6 col-lg-7 order-md-last">
8. <h4 **class**="d-flex justify-content-between align-items-center mb-3">
9. <span **class**="text-primary">Giỏ hàng</span> <span
10. **class**="badge bg-warning rounded-pill">${sessionScope.cart == **null** ? '0' : sessionScope.cart.tongsl}</span>
11. </h4>
12. <div **class**="table-responsive m-3">
13. <table **class**="table text-center">
14. <thead>
15. <tr>
16. <th scope="col" **class**="text-start h5">GIỎ HÀNG CỦA BẠN</th>
17. <th scope="col">Kích thước</th>
18. <th scope="col">Màu</th>
19. <th scope="col">Số lượng</th>
20. <th scope="col">Giá</th>
21. </tr>
22. </thead>
23. <tbody>
24. <!-- Start item -->
25. <c:forEach var="item" items="${sessionScope.cart.detail}"
26. varStatus="loop">
27. <tr id="tr-${item.value.id}">
28. <th scope="row">
29. <div **class**="d-flex align-items-center">
30. <div>
31. <img src="../images/${item.value.image}"
32. **class**="img-fluid rounded-3" style="width: 120px" />
33. </div>
34. <div **class**="flex-fill">
35. <div **class**="flex-column ms-4">
36. <p **class**="mb-2">${item.value.name}</p>
37. <p **class**="mb-0">
38. <fmt:formatNumber type="number" value="${item.value.price}" />
39. VND
40. </p>
41. </div>
42. </div>
43. </div>
44. </th>
45. <td **class**="align-middle">
46. <p **class**="mb-0" style="font-weight: 500">${item.value.size}</p>
47. </td>
48. <td **class**="align-middle">
49. <p **class**="mb-0" style="font-weight: 500">${item.value.color}</p>
50. </td>
51. <td **class**="align-middle">
52. <p **class**="mb-0" style="font-weight: 500">${item.value.quantityOder}</p>
53. </td>
54. <td **class**="align-middle" style="width: 180px">
55. <p **class**="mb-0" style="font-weight: 500"
56. id="totalSP-${cartDetail.value.masp}">
57. <fmt:formatNumber type="number"
58. value="${item.value.price \* item.value.quantityOder}" />
59. VND
60. </p>
61. </td>
62. </tr>
63. </c:forEach>
64. <!-- End item -->
65. </tbody>
66. </table>
67. <div **class**="d-flex justify-content-between mb-4 h2"
68. style="font-weight: 500">
69. <p **class**="mb-2">Tổng tiền:</p>
70. <p **class**="mb-2" id="tongTien">
71. <fmt:formatNumber type="number"
72. value="${sessionScope.cart.tongtien}" />
73. VND
74. </p>
75. </div>
76. </div>
77. </div>
78. <div **class**="col-md-6 col-lg-5">
79. <h4 **class**="mb-3">Địa chỉ thanh toán</h4>
80. <form **class**="needs-validation" action="/cart/paynow">
81. <div **class**="row g-3">
82. <div **class**="col-sm-12">
83. <label **for**="firstName" **class**="form-label">Tên đầy đủ</label> <input
84. type="text" **class**="form-control" id="fullname" name="fullname"
85. value="${sessionScope.user.fullname }" required="">
86. <div **class**="invalid-feedback">Tên hợp lệ là bắt buộc.</div>
87. </div>
89. <div **class**="col-12">
90. <label **for**="email" **class**="form-label">Email</label> <input
91. type="email" **class**="form-control" id="email" name="email"
92. placeholder="you@example.com" value="${sessionScope.user.email }">
93. <div **class**="invalid-feedback">Vui lòng nhập một địa chỉ
94. email hợp lệ để cập nhật thông tin vận chuyển.</div>
95. </div>
97. <div **class**="col-12">
98. <label **for**="address" **class**="form-label">Địa chỉ</label> <input
99. type="text" **class**="form-control" id="address" name="address"
100. placeholder="1234 Main St" required="">
101. <div **class**="invalid-feedback">Vui lòng nhập thông tin vận
102. chuyển của bạn địa chỉ.</div>
103. </div>
105. <div **class**="col-12">
106. <label **for**="phone" **class**="form-label">Số điện thoại</label> <input
107. type="text" **class**="form-control" id="phone" name="phone"
108. placeholder="01234567891" required="">
109. <div **class**="invalid-feedback">Vui lòng nhập số điện thoại
110. của bạn.</div>
111. </div>
113. </div>
115. <hr **class**="my-4">
117. <button **class**="w-100 btn btn-primary btn-lg" type="submit">Thanh
118. toán</button>
119. </form>
120. </div>
121. </div>
122. </div>

**ACCOUNT**

**Login**

1. <%@ page language="java" contentType="text/html; charset=UTF-8" pageEncoding="UTF-8" %>
2. <div **class**="d-flex align-items-center justify-content-center p-2">
3. <div **class**="col-12 col-sm-11 col-md-9 col-lg-7 col-xl-5 text-center p-2 shadow-lg rounded-3">
4. <form action="/account/login" method="post">
5. <h1 **class**="text-center">Đăng nhập</h1>
6. <div **class**="mb-3">
7. <input **class**="form-control" name="username" type="text" placeholder="Tên đăng nhập" />
8. </div>
9. <div **class**="mb-3">
10. <input **class**="form-control" name="password" type="password" placeholder="Mật khẩu" />
11. </div>
12. <div **class**="mb-1">
13. <a **class**="text-danger" href="#">Quên mật khẩu?</a>
14. <a href="/account/signup">Chưa có tài khoản? Đăng ký!</a>
15. </div>
16. <div **class**="mb-3">
17. <button **class**="btn btn-outline-primary">Đăng nhập</button>
18. </div>
19. </form>
20. <p><b>${message}${param.error}</b></p>
21. </div>
22. </div>

**Register**

1. <%@ page language="java" contentType="text/html; charset=UTF-8"
2. pageEncoding="UTF-8"%>
4. <div **class**=" d-flex align-items-center justify-content-center ">
5. <div **class**="col-12 col-sm-11 col-md-9 col-lg-7 col-xl-5 text-center p-2 shadow-lg rounded-3">
6. <form action="/account/signup" method="post">
7. <h1>Đăng ký</h1>
8. <div **class**="mb-3">
9. <input **class**="form-control" type="text" name="username" placeholder="Tên đăng nhập" />
10. </div>
11. <div **class**="mb-3">
12. <input **class**="form-control" type="text" name="fullname" placeholder="Tên đầy đủ" />
13. </div>
14. <div **class**="mb-3">
15. <input **class**="form-control" type="email" name="email" placeholder="Email" />
16. </div>
17. <div **class**="mb-3">
18. <input **class**="form-control" type="password" name="password" placeholder="Mật khẩu" />
19. </div>
21. <div **class**="mb-1">
22. <a href="/account/login">Có tài khoản! Đăng nhập!</a>
23. </div>
24. <div **class**="mb-3">
25. <button **class**="btn btn-outline-success">Đăng ký</button>
26. </div>
27. </form>
28. <p><b><i>${message}${param.error}</i></b></p>
29. </div>
30. </div>
31. **Back-end**

**XML**

1. **<?xml** version="1.0" encoding="UTF-8"**?>**
2. **<project** xmlns="http://maven.apache.org/POM/4.0.0"
3. xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
4. xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 https://maven.apache.org/xsd/maven-4.0.0.xsd"**>**
5. **<modelVersion>**4.0.0**</modelVersion>**
6. **<parent>**
7. **<groupId>**org.springframework.boot**</groupId>**
8. **<artifactId>**spring-boot-starter-parent**</artifactId>**
9. **<version>**2.6.8**</version>**
10. **<relativePath** **/>** <!-- lookup parent from repository -->
11. **</parent>**
12. **<groupId>**com.poly**</groupId>**
13. **<artifactId>**WebShop-ASM-Java5**</artifactId>**
14. **<version>**0.0.1-SNAPSHOT**</version>**
15. **<packaging>**war**</packaging>**
16. **<name>**WebShop-ASM-Java5**</name>**
17. **<description>**Demo project for Spring Boot**</description>**
18. **<properties>**
19. **<java.version>**1.8**</java.version>**
20. **</properties>**
21. **<dependencies>**
22. **<dependency>**
23. **<groupId>**org.springframework.boot**</groupId>**
24. **<artifactId>**spring-boot-starter-web**</artifactId>**
25. **</dependency>**
27. **<dependency>**
28. **<groupId>**org.projectlombok**</groupId>**
29. **<artifactId>**lombok**</artifactId>**
30. **<optional>**true**</optional>**
31. **</dependency>**
32. **<dependency>**
33. **<groupId>**org.springframework.boot**</groupId>**
34. **<artifactId>**spring-boot-starter-tomcat**</artifactId>**
35. **<scope>**provided**</scope>**
36. **</dependency>**
37. **<dependency>**
38. **<groupId>**org.springframework.boot**</groupId>**
39. **<artifactId>**spring-boot-starter-test**</artifactId>**
40. **<scope>**test**</scope>**
41. **</dependency>**
42. <!-- Apache Tomcat -->
43. **<dependency>**
44. **<groupId>**org.apache.tomcat.embed**</groupId>**
45. **<artifactId>**tomcat-embed-jasper**</artifactId>**
46. **</dependency>**
47. <!-- https://mvnrepository.com/artifact/javax.servlet/jstl -->
48. **<dependency>**
49. **<groupId>**javax.servlet**</groupId>**
50. **<artifactId>**jstl**</artifactId>**
51. **</dependency>**
52. <!-- https://mvnrepository.com/artifact/org.springframework.boot/spring-boot-starter-validation -->
53. **<dependency>**
54. **<groupId>**org.springframework.boot**</groupId>**
55. **<artifactId>**spring-boot-starter-validation**</artifactId>**
56. **</dependency>**
57. <!-- spring-boot-starter-data-jpa -->
58. **<dependency>**
59. **<groupId>**org.springframework.boot**</groupId>**
60. **<artifactId>**spring-boot-starter-data-jpa**</artifactId>**
61. **</dependency>**
62. <!-- mssql-jdbc -->
63. **<dependency>**
64. **<groupId>**com.microsoft.sqlserver**</groupId>**
65. **<artifactId>**mssql-jdbc**</artifactId>**
66. **<scope>**runtime**</scope>**
67. **</dependency>**
68. **<dependency>**
69. **<groupId>**org.apache.tiles**</groupId>**
70. **<artifactId>**tiles-jsp**</artifactId>**
71. **<version>**3.0.8**</version>**
72. **</dependency>**
73. **<dependency>**
74. **<groupId>**org.springframework.boot**</groupId>**
75. **<artifactId>**spring-boot-starter-mail**</artifactId>**
76. **</dependency>**
77. **</dependencies>**
79. **<build>**
80. **<plugins>**
81. **<plugin>**
82. **<groupId>**org.springframework.boot**</groupId>**
83. **<artifactId>**spring-boot-maven-plugin**</artifactId>**
84. **<configuration>**
85. **<excludes>**
86. **<exclude>**
87. **<groupId>**org.projectlombok**</groupId>**
88. **<artifactId>**lombok**</artifactId>**
89. **</exclude>**
90. **</excludes>**
91. **</configuration>**
92. **</plugin>**
93. **</plugins>**
94. **</build>**
96. **</project>**

**Properties setup**

1. spring.mvc.view.prefix= /WEB-INF/views/
2. spring.mvc.view.suffix= .jsp
3. spring.servlet.multipart.max-file-size= 20MB
4. spring.servlet.multipart.max-request-size= 20MB
6. spring.datasource.url=jdbc:sqlserver://localhost;database=ShopAoNhom1
7. spring.datasource.username=sa
8. spring.datasource.password=123
9. spring.datasource.driverClassName=com.microsoft.sqlserver.jdbc.SQLServerDriver
10. spring.jpa.hibernate.dialect=org.hibernate.dialect.SQLServer2012Dialect
11. spring.jpa.show-sql=false
12. spring.jpa.hibernate.ddl-auto = none
14. spring.mail.host=smtp.gmail.com
15. spring.mail.port=587
16. spring.mail.username=phathtps15403@fpt.edu.vn
17. spring.mail.password=ma1412pet
18. spring.mail.properties.mail.smpt.auth=true
19. spring.mail.properties.mail.smtp.starttls.enable=true

**Com.poly**

**InterConfig**

1. **package** com.poly;
3. **import** org.springframework.beans.factory.annotation.Autowired;
4. **import** org.springframework.context.annotation.Configuration;
5. **import** org.springframework.web.servlet.config.annotation.InterceptorRegistry;
6. **import** org.springframework.web.servlet.config.annotation.WebMvcConfigurer;
8. **import** com.poly.interceptor.AuthInterceptor;
9. **import** com.poly.interceptor.GlobalInterceptor;


13. @Configuration
14. **public** **class** InterConfig **implements** WebMvcConfigurer{
15. @Autowired
16. GlobalInterceptor global;
18. @Autowired
19. AuthInterceptor auth;
21. @Override
22. **public** **void** addInterceptors(InterceptorRegistry registry) {
23. registry.addInterceptor(global)
24. .addPathPatterns("/\*\*")
25. .excludePathPatterns("/assets/\*\*");
27. registry.addInterceptor(auth)
28. .addPathPatterns("/account/edit", "/account/chgpwd", "/order/\*\*"
29. , "/admin/\*\*"
30. )
31. .excludePathPatterns("/assets/\*\*", "/admin/home/index");
32. }
33. }

**TilesConfig**

1. **package** com.poly;
3. **import** org.springframework.context.annotation.Bean;
4. **import** org.springframework.context.annotation.Configuration;
5. **import** org.springframework.web.servlet.ViewResolver;
6. **import** org.springframework.web.servlet.view.UrlBasedViewResolver;
7. **import** org.springframework.web.servlet.view.tiles3.TilesConfigurer;
8. **import** org.springframework.web.servlet.view.tiles3.TilesView;
10. @Configuration
11. **public** **class** TilesConfig {
12. @Bean("viewResolver")
13. **public** ViewResolver getViewResolver() {
14. UrlBasedViewResolver viewResolver = **new** UrlBasedViewResolver();
15. viewResolver.setViewClass(TilesView.**class**);
16. **return** viewResolver;
17. }
19. @Bean("tilesConfigurer")
20. **public** TilesConfigurer getTilesConfigurer() {
21. TilesConfigurer tilesConfigurer = **new** TilesConfigurer();
22. tilesConfigurer.setDefinitions("/WEB-INF/views/layout/tiles.xml");
23. **return** tilesConfigurer;
24. }
25. }

**Com.poly.controller**

**Account**

1. @Controller
2. **public** **class** AccountController {
3. @Autowired
4. AccountDAO dao;
5. @Autowired
6. SessionService session;
7. @Autowired
8. HttpServletRequest request;
10. @GetMapping("/account/login")
11. **public** String login() {
12. **return** "account/login";
13. }
15. @PostMapping("/account/login")
16. **public** String login(Model model, @RequestParam("username") String username,
17. @RequestParam("password") String password) {
18. **try** {
19. Account user = dao.getById(username);
20. **if** (!user.getPassword().equals(password)) {
21. session.set("ktdn", **false**);
22. model.addAttribute("message", "Sai tài khoản hoặc mật khẩu!");
23. } **else** {
24. session.set("user", user);
25. session.set("ktdn", **true**);
26. String uri = session.get("security-uri");
27. **if** (uri != **null**) {
28. **return** "redirect:" + uri;
29. } **else** {
30. model.addAttribute("message", "Đăng nhập thành công!");
31. **return** "redirect:/home/index";
32. }
33. }
34. } **catch** (Exception e) {
35. session.set("ktdn", **false**);
36. model.addAttribute("message", "Sai tài khoản hoặc mật khẩu!");
37. }
38. **return** "account/login";
39. }
41. @GetMapping("/account/signup")
42. **public** String signup() {
43. **return** "account/signup";
44. }
46. @PostMapping("/account/signup")
47. **public** String signup(Model model, @RequestParam("username") String username, @RequestParam("email") String email,
48. @RequestParam("password") String password, @RequestParam("fullname") String fullname) {
49. **boolean** kt = **true**;
50. **for** (Account acc : dao.findAll()) {
51. **if** (acc.getUsername().equalsIgnoreCase(username)) {
52. kt = **false**;
53. **break**;
54. }
55. }
56. **if** (kt) {
57. Account newAcc = **new** Account(username, password, fullname, email, "", **false**, **true**, **null**);
58. dao.save(newAcc);
59. model.addAttribute("message", "Đăng ký Thành công!");
60. } **else** {
61. model.addAttribute("message", "Username đã tồn tại!");
62. }
63. **return** "account/login";
64. }
66. @RequestMapping("/account/logout")
67. **public** String logout(Model model) {
68. session.set("user", **null**);
69. session.set("ktdn", **null**);
70. **return** "redirect:/home/index";
71. }
73. }

**Admin**

1. @Controller
2. @RequestMapping("/admin")
3. **public** **class** AdminController {
4. @Autowired
5. ProductDAO prodao;
6. @Autowired
7. BrandDAO bradao;
8. @Autowired
9. CategoryDAO catdao;
10. @Autowired
11. SizeDAO sizedao;
12. @Autowired
13. ColorDAO colordao;
14. @Autowired
15. ImageDAO imageDAO;
16. @Autowired
17. ServletContext app;
18. @Autowired
19. HttpServletRequest request;
20. @Autowired
21. ParamService param;
22. @Autowired
23. PageShow pageShow;
25. @RequestMapping("/index")
26. **public** String index() {
27. request.setAttribute("pageName", "Dashboard");
28. **return** "admin/index";
29. }
31. @RequestMapping("/product-list")
32. **public** String viewProductList(@RequestParam("p") Optional<Integer> p, Model model,
33. @RequestParam("field") Optional<String> field) {
34. pageShow.pageProductAll(p, field, model, "pageproduct");
35. request.setAttribute("pageName", "Product Management");
36. **return** "admin/product-list";
37. }
39. @ModelAttribute("brands")
40. **public** List<Brand> getBrand() {
41. **return** bradao.findAll();
42. }
44. @ModelAttribute("categorys")
45. **public** List<Category> getCategory() {
46. **return** catdao.findAll();
47. }
49. @RequestMapping("/product-editform")
50. **public** String viewProductEditForm(Model model) {
51. model.addAttribute("prod", **new** Product());
52. request.setAttribute("pageName", "Product Management");
53. **return** "admin/product-editform";
54. }
56. @RequestMapping("/product-edit")
57. **public** String edit(Model model, @RequestParam("id") String id) {
58. request.setAttribute("pageName", "Product Management");
59. Product prod = prodao.getById(id);
60. model.addAttribute("prod", prod);
61. **return** "admin/product-editform";
62. }
64. @RequestMapping("/product-save")
65. **public** String saveProduct(Product product) {
66. prodao.save(product);
67. **return** "redirect:/admin/product-list";
68. }
70. @RequestMapping("/product-update")
71. **public** String updateProduct(Product product) {
72. prodao.save(product);
73. **return** "redirect:/admin/product-list";
74. }
76. @RequestMapping("/product-delete")
77. **public** String deleteProduct(@RequestParam("id") String id) {
78. prodao.deleteById(id);
79. **return** "redirect:/admin/product-list";
80. }
82. // size edit
84. @RequestMapping("/size-seachid")
85. **public** String viewSizeSeachId(@RequestParam("p") Optional<Integer> p, Model model,
86. @RequestParam("field") Optional<String> field, @RequestParam("idproseach") String id) {
87. request.setAttribute("pageName", "Size Management");
88. pageShow.pageProductSeach(p, field, model, "pagesize", id);
89. **return** "admin/size-editform";
90. }
92. @RequestMapping("/size-editform")
93. **public** String viewSizeEditForm(@RequestParam("p") Optional<Integer> p, Model model,
94. @RequestParam("field") Optional<String> field) {
95. request.setAttribute("pageName", "Size Management");
96. pageShow.pageProductAll(p, field, model, "pagesize");
97. **return** "admin/size-editform";
98. }
100. @RequestMapping("/size-edit")
101. **public** String editSize(@RequestParam("p") Optional<Integer> p, @RequestParam("field") Optional<String> field,
102. Model model, @RequestParam("id") String id) {
103. request.setAttribute("pageName", "Size Management");
104. model.addAttribute("idprod", id);
105. pageShow.pageProductAll(p, field, model, "pagesize");
106. List<String> list = **new** ArrayList<>();
107. **for** (Size size : prodao.getById(id).getSizes()) {
108. list.add(size.getName());
109. }
110. **return** "admin/size-editform";
111. }
113. @RequestMapping("/size-save")
114. **public** String saveSize(@RequestParam("idprod") String id, @RequestParam("name") List<String> list) {
115. Product pro = prodao.getById(id);
116. **if** (!pro.getSizes().isEmpty()) {
117. **for** (Size s : pro.getSizes()) {
118. sizedao.delete(s);
119. }
120. }
121. **for** (String string : list) {
122. sizedao.save(**new** Size(string, pro));
123. }
124. **return** "redirect:/admin/size-editform";
125. }
127. // image edit

130. @RequestMapping("/image-editform")
131. **public** String viewImageEditForm(@RequestParam("p") Optional<Integer> p, Model model,
132. @RequestParam("field") Optional<String> field) {
133. request.setAttribute("pageName", "Image Management");
134. pageShow.pageImageAll(p, field, model, "page");
135. **return** "admin/image-editform";
136. }
138. @RequestMapping("/image-seachid")
139. **public** String viewImageSeachId(@RequestParam("p") Optional<Integer> p, Model model,
140. @RequestParam("field") Optional<String> field, @RequestParam("idimgseach") String id) {
141. request.setAttribute("pageName", "Image Management");
142. pageShow.pageImageSeach(p, field, model, "page", id);
143. **return** "admin/image-editform";
144. }

147. // color edit
148. @RequestMapping("/color-editform")
149. **public** String viewColorEditForm(Model model) {
150. model.addAttribute("prod", **new** Product());
151. request.setAttribute("pageName", "Color Management");
152. **return** "admin/color-editform";
153. }
154. }

**Product**

1. @Controller
2. @RequestMapping("/product")
3. **public** **class** ProductContronller {
4. @Autowired
5. ProductDAO dao;
7. @RequestMapping("/detail")
8. **public** String detail(Model model, @RequestParam("id") String id) {
9. Product product = dao.getById(id);
10. model.addAttribute("item", product);
11. **return** "product/detail";
12. }

15. }

**Shoppingcart**

1. @Controller
2. @RequestMapping("/cart")
3. **public** **class** CartController {
4. @Autowired
5. ParamService param;
6. @Autowired
7. SessionService sessions;
8. @Autowired
9. HttpServletRequest req;
10. @Autowired
11. ShoppingCartService cart;
12. @Autowired
13. HttpSession session;
14. @Autowired
15. CartService cartService;
16. @Autowired
17. MailerService mailer;
19. @RequestMapping("/view")
20. **public** String viewCart() {
21. **return** "cartshop/shoppingcart";
22. }
24. @GetMapping("/add/{id}/{isUpdate}")
25. **public** String addToCart(@PathVariable("id") String id, @PathVariable("isUpdate") **boolean** isUpdate) {
26. CartDto cart = (CartDto) session.getAttribute("cart");
27. //      // update Cart
28. String size, color;
29. **if** (param.getString("size", "").equals("")) {
30. size = "S (Mặc định)";
31. } **else** {
32. size = param.getString("size", "");
33. }
34. **if** (param.getString("color", "").equals("")) {
35. color = "White";
36. } **else** {
37. color = param.getString("color", "");
38. }
39. session.setAttribute("cart",
40. cartService.updateCart(cart, "newsanpham", id, param.getInt("soLuongMua", 0), isUpdate, size, color));
41. **return** "redirect:/home/index";
42. }
44. @RequestMapping("/update/{id}/{isUpdate}")
45. **public** String updateToCart(@PathVariable("id") String id, @PathVariable("isUpdate") **boolean** isUpdate,
46. @RequestParam("soLuongMua") **int** soluong) {
47. CartDto cart = (CartDto) session.getAttribute("cart");
48. HashMap<String, CartDetailDto> cartDetail = cart.getDetail();
49. cartDetail.forEach((key, detail) -> {
50. **if** (detail.getId().equals(id)) {
51. session.setAttribute("cart",
52. cartService.updateCart(cart, key, id, soluong, isUpdate, detail.getSize(), detail.getColor()));
53. }
54. });
55. **return** "redirect:/cart/view";
56. }
58. @GetMapping("/remove/{id}")
59. **public** String removeFromCart(@PathVariable String id) {
60. CartDto cart = (CartDto) session.getAttribute("cart");
61. HashMap<String, CartDetailDto> cartDetail = cart.getDetail();
62. List<String> list = **new** ArrayList<>();
63. cartDetail.forEach((key, detail) -> {
64. **if** (detail.getId().equals(id)) {
65. list.add(key);
66. }
67. });
68. **for** (String key : list) {
69. cartDetail.remove(key);
70. }
71. cart.setTongsl(cartService.getTongSL(cart));
72. cart.setTongtien(cartService.getTongTien(cart));
73. **return** "redirect:/cart/view";
74. }
76. @GetMapping("/clear")
77. **public** String clearCart() {
78. CartDto cart = (CartDto) session.getAttribute("cart");
79. HashMap<String, CartDetailDto> cartDetail = cart.getDetail();
80. List<String> list = **new** ArrayList<>();
81. cartDetail.forEach((key, detail) -> {
82. list.add(key);
83. });
84. **for** (String key : list) {
85. cartDetail.remove(key);
86. }
87. cart.setTongsl(cartService.getTongSL(cart));
88. cart.setTongtien(cartService.getTongTien(cart));
89. **return** "redirect:/cart/view";
90. }
92. @RequestMapping("/checkout")
93. **public** String viewCheckout(Model model) {
94. Account user = (Account) session.getAttribute("user");
95. **if** (user != **null**) {
97. **return** "cartshop/checkout";
98. } **else** {
99. model.addAttribute("message", "Vui lòng đăng nhập!");
100. **return** "redirect:/account/login";
101. }
103. }
105. @GetMapping("/paynow")
106. **public** String doGetPayNow(@RequestParam("phone") String phone, @RequestParam("address") String address,
107. RedirectAttributes redirectAttributes, HttpSession session) {
108. CartDto cart = (CartDto) session.getAttribute("cart");
109. **if** (cart.getTongsl() == 0) {
110. redirectAttributes.addFlashAttribute("error", "Giỏ hàng rỗng, vui lòng mua hàng");
111. **return** "redirect:/cart/view";
112. } **else** {
113. Account user = (Account) session.getAttribute("user");
114. cart.setUserId(user.getUsername());
115. cart.setDiachi(address);
116. cart.setDienthoai(phone);
117. cartService.insertCart(cart);
118. session.setAttribute("cart", **new** CartDto());
119. **try** {
120. mailer.send(user.getEmail(), "Đặt Hàng Thành Công",
121. "Chào " + user.getFullname() + " đơn hàng của bạn đặt thành công.\n" + "Địa chỉ nhận hàng: "
122. + address + "\n" + "Số điện thoại: " + phone + "\n" + "Tổng tiền Đơn hàng: "
123. + cart.getTongtien());
124. } **catch** (MessagingException e) {
125. }
126. **return** "redirect:/home/index";
128. }
129. }
130. }

**Home**

1. @Controller
2. **public** **class** HomeController {
3. @Autowired
4. HttpServletRequest request;
5. @Autowired
6. ProductDAO dao;
7. @Autowired
8. CategoryDAO daoca;
9. @Autowired
10. BrandDAO daobr;
11. @Autowired
12. SessionService session;
14. @RequestMapping(value = { "/home/index", "/" })
15. **public** String index(@RequestParam("p") Optional<Integer> p) {
16. **if** (session.get("ktdn") == **null**) {
17. session.set("ktdn", **false**);
18. }
19. List<Category> categories = daoca.findAll();
20. request.setAttribute("categories", categories);
21. List<Brand> brands = daobr.findAll();
22. request.setAttribute("brands", brands);
23. Pageable pageable = PageRequest.of(p.orElse(0), 8);
24. List<Product> list = dao.findAll();
25. **for** (Product product : list) {
26. **if** (product.getColors().size() == 0) {
27. product.setAvailable(**false**);
28. dao.save(product);
29. }
30. }
31. Page<Product> page = dao.findAllByAvailableTrue(pageable);
32. request.setAttribute("page", page);
34. **return** "home/index";
35. }
37. //  @RequestMapping("/home/about")
38. //  public String about() {
39. //      return "home/about";
40. //  }
41. }

**Com.poly.bean**

**Mail**

1. @Data
2. @NoArgsConstructor
3. @AllArgsConstructor
4. **public** **class** MailInfo {
5. String from;
6. String to;
7. String[] cc;
8. String[] bcc;
9. String subject;
10. String body;
11. String[] attachments;
12. **public** MailInfo(String to, String subject, String body) {
13. **super**();
14. **this**.from = "FPT Polytechnic <poly@fpt.edu.vn>";
15. **this**.to = to;
16. **this**.subject = subject;
17. **this**.body = body;
18. }
19. }

**Com.poly.filter**

**CartFilter**

1. @Component
2. **public** **class** CartFilter **implements** Filter {
4. @Override
5. **public** **void** doFilter(ServletRequest request, ServletResponse response, FilterChain chain)
6. **throws** IOException, ServletException {
7. HttpServletRequest httpServletReq = (HttpServletRequest) request;
8. HttpSession session = httpServletReq.getSession();
9. CartDto cart = (CartDto) session.getAttribute("cart");
10. **if** (cart == **null**) {
11. session.setAttribute("cart", **new** CartDto());
12. }
13. chain.doFilter(request, response);
14. }
15. }

**Com.poly.interceptor**

**AuthInterceptor**

1. @Service
2. **public** **class** AuthInterceptor **implements** HandlerInterceptor {
3. @Autowired
4. SessionService session;
6. @Override
7. **public** **boolean** preHandle(HttpServletRequest request, HttpServletResponse response, Object handler)
8. **throws** Exception {
9. String uri = request.getRequestURI();
10. Account user = session.get("user");
12. String error = "";
13. **if** (user == **null**) {
14. error = "Vui lòng đăng nhập!";
15. } **else** **if** (!user.isAdmin() && uri.startsWith("/admin/")) {
16. error = "Truy cập bị từ chối!";
17. }
18. **if** (error.length() > 0) {
19. session.set("security-uri", uri);
20. response.sendRedirect("/account/login?error=" + URLEncoder.encode(error, "UTF-8"));
21. **return** **false**;
22. }
23. **return** **true**;
24. }
25. }

**GlobalInterceptor**

1. @Service
2. **public** **class** GlobalInterceptor **implements** HandlerInterceptor{
3. @Autowired
4. CategoryDAO dao;
6. @Override
7. **public** **boolean** preHandle(HttpServletRequest request, HttpServletResponse response, Object handler)
8. **throws** Exception {
9. request.setAttribute("uri", request.getRequestURI());
10. **return** **true**;
11. }
13. @Override
14. **public** **void** postHandle(HttpServletRequest request, HttpServletResponse response, Object handler,
15. ModelAndView modelAndView) **throws** Exception {
16. request.setAttribute("categories", dao.findAll());
17. }
18. }

**Com.poly.service**

**CartService**

1. @Service
2. **public** **class** CartService {
4. @Autowired
5. **private** OrderDAO odao;
7. @Autowired
8. **private** OrderDetailDAO oddao;
10. @Autowired
11. **private** ProductDAO prdao;
13. @Autowired
14. **private** AccountDAO accdao;
16. @Transactional
17. **public** **void** insertCart(CartDto cart) {
18. Order hoadon = mappingDtoToOrderEntity(cart);
19. **try** {
20. // insert HoaDon
21. Order hdReturn = odao.saveAndFlush(hoadon);
22. List<OrderDetail> listHDCT = mappingDtoToOrderDetailEntity(cart.getDetail(), hdReturn);
23. // insert HDCT
24. listHDCT.forEach(hdct -> {
25. oddao.saveAndFlush(hdct);
27. // update SL cua SP sau khi insert HDCT
28. Product sp = prdao.getById(hdct.getOderproduct().getId());
29. **int** newQuantity = sp.getQuantity() - hdct.getQuantity();
30. sp.setQuantity(newQuantity);
31. prdao.saveAndFlush(sp);
32. });
33. } **catch** (Exception ex) {
34. ex.printStackTrace();
35. }
36. }
38. **public** CartDto updateCart(CartDto cart,String key, String masp, **int** soLuong, **boolean** isUpdate, String size, String color) {
39. Product product = prdao.getById(masp);
41. HashMap<String, CartDetailDto> cartDetail = cart.getDetail();
42. **if** (cartDetail.containsKey(key)) {
43. **if** (soLuong < 1) {
44. cartDetail.remove(key);
45. } **else** {
46. **if** (isUpdate) {
47. cartDetail.get(key).setQuantityOder(soLuong);
48. } **else** {
49. **if** (cartDetail.get(key).getSize().equals(size) && cartDetail.get(key).getColor().equals(color)) {
50. **int** slCu = cartDetail.get(key).getQuantityOder();
51. cartDetail.get(key).setQuantityOder(slCu + soLuong);
52. } **else** {
53. addToCart(masp, product, size, color, soLuong, cartDetail);
54. }
55. }
56. }
58. } **else** {
59. addToCart(masp, product, size, color, soLuong, cartDetail);
60. }
62. cart.setTongsl(**this**.getTongSL(cart));
63. cart.setTongtien(**this**.getTongTien(cart));
65. **return** cart;
66. }
68. **private** **void** addToCart(String masp, Product product, String size, String color, **int** soLuong,
69. HashMap<String, CartDetailDto> cartDetail) {
70. CartDetailDto cartDetailDto = **new** CartDetailDto();
71. cartDetailDto.setName(product.getName());
72. cartDetailDto.setPrice(product.getPrice());
73. cartDetailDto.setId(product.getId());
74. cartDetailDto.setSize(size);
75. cartDetailDto.setColor(color);
76. String image = "";
77. **for** (Color c : product.getColors()) {
78. **if** (c.getName().equals(color)) {
79. image = c.getImage().getName();
80. **break**;
81. }
82. }
83. cartDetailDto.setImage(image);
84. cartDetailDto.setQuantityOder(soLuong);
85. cartDetail.put(masp + (cartDetail.size() + 1), cartDetailDto);
87. }
89. **public** **double** getTongTien(CartDto cart) {
90. HashMap<String, CartDetailDto> cartDetail = cart.getDetail();
91. Double tongTien = 0D;
92. **for** (CartDetailDto cartDetailDto : cartDetail.values()) {
93. Product product = prdao.getById(cartDetailDto.getId());
94. tongTien += product.getPrice() \* cartDetailDto.getQuantityOder();
95. }
96. **return** tongTien;
97. }
99. **public** **int** getTongSL(CartDto cart) {
100. HashMap<String, CartDetailDto> cartDetail = cart.getDetail();
101. **int** tongSL = 0;
102. **for** (CartDetailDto productDto : cartDetail.values()) {
103. tongSL += productDto.getQuantityOder();
104. }
105. **return** tongSL;
106. }
108. **private** Order mappingDtoToOrderEntity(CartDto cartDto) {
109. Order hoadon = **new** Order();
110. hoadon.setAddress(cartDto.getDiachi());
111. hoadon.setPhone(cartDto.getDienthoai());
112. hoadon.setTotalmoney(cartDto.getTongtien());
113. Optional<Account> user = accdao.findById(cartDto.getUserId());
114. hoadon.setAccount(user.get());
115. **return** hoadon;
116. }
117. **private** List<OrderDetail> mappingDtoToOrderDetailEntity(HashMap<String, CartDetailDto> details, Order hoadon) {
118. List<OrderDetail> result = **new** ArrayList<>();
119. details.forEach((key, hdct) -> {
120. OrderDetail hoadonchitiet = **new** OrderDetail();
121. Product sanpham = prdao.getById(hdct.getId());
123. hoadonchitiet.setOderproduct(sanpham);
124. hoadonchitiet.setPrice(hdct.getPrice());
125. hoadonchitiet.setQuantity(hdct.getQuantityOder());
126. hoadonchitiet.setOrder(hoadon);
127. **for** (Color c : sanpham.getColors()) {
128. **if** (c.getName().equals(hdct.getColor())) {
129. hoadonchitiet.setColor(c);
130. }
131. }
132. **for** (Size s : sanpham.getSizes()) {
133. **if** (s.getName().equals(hdct.getSize())) {
134. hoadonchitiet.setSize(s);
135. }
136. }
137. result.add(hoadonchitiet);
138. });
139. **return** result;
140. }
141. }

**CookieService**

1. @Service
2. **public** **class** CookieService {
3. @Autowired
4. HttpServletRequest request;
5. @Autowired
6. HttpServletResponse response;
8. /\*\*
9. \* Đọc cookie từ request
10. \*
11. \* @param name tên cookie cần đọc
12. \* @return đối tượng cookie đọc được hoặc null nếu không tồn tại
13. \*/
14. **public** Cookie get(String name) {
15. Cookie[] cookies = request.getCookies();
16. **if** (cookies != **null**) {
17. **for** (Cookie cookie : cookies) {
18. **if** (cookie.getName().equalsIgnoreCase(name)) {
19. **return** cookie;
20. }
21. }
22. }
23. **return** **null**;
24. }
26. /\*\*
27. \* Đọc giá trị của cookie từ request
28. \*
29. \* @param name tên cookie cần đọc
30. \* @return chuỗi giá trị đọc được hoặc rỗng nếu không tồn tại
31. \*/
32. **public** String getValue(String name) {
33. Cookie[] cookies = request.getCookies();
34. **if** (cookies != **null**) {
35. **for** (Cookie cookie : cookies) {
36. **if** (cookie.getName().equalsIgnoreCase(name)) {
37. **return** cookie.getValue();
38. }
39. }
40. }
41. **return** **null**;
42. }
44. /\*\*
45. \* Tạo và gửi cookie về client
46. \*
47. \* @param name  tên cookie
48. \* @param value giá trị cookie
49. \* @param hours thời hạn (giờ)
50. \* @return đối tượng cookie đã tạo
51. \*/
52. **public** **void** add(String name, String value, **int** days) {
53. Cookie cookie = **new** Cookie(name, value);
54. cookie.setMaxAge(days \* 60 \* 60);
55. cookie.setPath("/");
56. response.addCookie(cookie);
57. }
59. /\*\*
60. \* Xóa cookie khỏi client
61. \*
62. \* @param name tên cookie cần xóa
63. \*/
64. **public** **void** remove(String name) {
65. Cookie[] cookies = request.getCookies();
66. **if** (cookies != **null**) {
67. **for** (Cookie cookie : cookies) {
68. **if** (cookie.getName().equalsIgnoreCase(name)) {
69. cookie.setMaxAge(0);
70. cookie.setPath("/");
71. response.addCookie(cookie);
72. **break**;
74. }
75. }
76. }
77. }
78. }

**MailerService**

1. @Service
2. **public** **class** MailerServiceImpl **implements** MailerService {
3. @Autowired
4. JavaMailSender sender;
6. @Override
7. **public** **void** send(MailInfo mail) **throws** MessagingException {
8. MimeMessage message = sender.createMimeMessage();
9. MimeMessageHelper helper = **new** MimeMessageHelper(message, **true**, "utf-8");
10. helper.setFrom(mail.getFrom());
11. helper.setTo(mail.getTo());
12. helper.setSubject(mail.getSubject());
13. helper.setText(mail.getBody(), **true**);
14. helper.setReplyTo(mail.getFrom());
16. String[] cc = mail.getCc();
17. **if** (cc != **null** && cc.length > 0) {
18. helper.setCc(cc);
19. }
20. String[] bcc = mail.getBcc();
21. **if** (bcc != **null** && bcc.length > 0) {
22. helper.setBcc(bcc);
23. }
24. String[] attachments = mail.getAttachments();
25. **if** (attachments != **null** && attachments.length > 0) {
26. **for** (String path : attachments) {
27. File file = **new** File(path);
28. helper.addAttachment(file.getName(), file);
29. }
30. }
31. sender.send(message);
32. }
34. @Override
35. **public** **void** send(String to, String subject, String body) **throws** MessagingException {
36. **this**.send(**new** MailInfo(to, subject, body));
37. }
39. List<MailInfo> queue = **new** ArrayList<>();
41. @Override
42. **public** **void** queue(MailInfo mail) {
43. queue.add(mail);
44. }
46. @Override
47. **public** **void** queue(String to, String subject, String body) {
48. queue(**new** MailInfo(to, subject, body));
49. }
51. @Scheduled(fixedDelay = 5000)
52. **public** **void** run() {
53. **while** (!queue.isEmpty()) {
54. MailInfo mail = queue.remove(0);
55. **try** {
56. send(mail);
57. } **catch** (Exception e) {
58. e.printStackTrace();
59. }
60. }
61. }
62. }

**ParamService**

1. @Service
2. **public** **class** ParamService {
3. @Autowired
4. HttpServletRequest request;
5. ServletContext app;
7. /\*\*
8. \* Đọc chuỗi giá trị của tham số
9. \*
10. \* @param name         tên tham số
11. \* @param defaultValue giá trị mặc định
12. \* @return giá trị tham số hoặc giá trị mặc định nếu không tồn tại
13. \*/
14. **public** String getString(String name, String defaultValue) {
15. **if** (request.getParameter(name) != **null**) {
16. **return** request.getParameter(name);
17. }
18. **return** defaultValue;
19. }
21. /\*\*
22. \* Đọc số nguyên giá trị của tham số
23. \*
24. \* @param name         tên tham số
25. \* @param defaultValue giá trị mặc định
26. \* @return giá trị tham số hoặc giá trị mặc định nếu không tồn tại
27. \*/
28. **public** **int** getInt(String name, **int** defaultValue) {
29. **if** (request.getParameter(name) != **null**) {
30. **return** Integer.parseInt(request.getParameter(name));
31. }
32. **return** defaultValue;
33. }
35. /\*\*
36. \* Đọc số thực giá trị của tham số
37. \*
38. \* @param name         tên tham số
39. \* @param defaultValue giá trị mặc định
40. \* @return giá trị tham số hoặc giá trị mặc định nếu không tồn tại
41. \*/
42. **public** **double** getDouble(String name, **double** defaultValue) {
43. **if** (request.getParameter(name) != **null**) {
44. **return** Double.parseDouble(request.getParameter(name));
45. }
46. **return** defaultValue;
47. }
49. /\*\*
50. \* Đọc giá trị boolean của tham số
51. \*
52. \* @param name         tên tham số
53. \* @param defaultValue giá trị mặc định
54. \* @return giá trị tham số hoặc giá trị mặc định nếu không tồn tại
55. \*/
56. **public** **boolean** getBoolean(String name, **boolean** defaultValue) {
57. **if** (request.getParameter(name) != **null**) {
58. **return** Boolean.parseBoolean(request.getParameter(name));
59. }
60. **return** defaultValue;
61. }
63. /\*\*
64. \* Đọc giá trị thời gian của tham số
65. \*
66. \* @param name    tên tham số
67. \* @param pattern là định dạng thời gian
68. \* @return giá trị tham số hoặc null nếu không tồn tại
69. \* @throws RuntimeException lỗi sai định dạng
70. \*/
71. **public** Date getDate(String name, String pattern) {
72. **if** (request.getParameter(name) != **null**) {
73. **try** {
74. SimpleDateFormat formatter = **new** SimpleDateFormat(pattern);
75. **return** formatter.parse(request.getParameter(name));
76. } **catch** (Exception e) {
77. **throw** **new** RuntimeException(e);
78. }
79. }
80. **return** **null**;
81. }
83. **public** File save(MultipartFile file, String path) {
84. File dir = **new** File(app.getRealPath(path));
85. **if** (!dir.exists()) {
86. dir.mkdir();
87. }
88. **try** {
89. File saveFile = **new** File(dir, file.getOriginalFilename());
90. file.transferTo(saveFile);
91. **return** saveFile;
92. } **catch** (Exception e) {
93. **throw** **new** RuntimeException(e);
94. }
95. }
96. }

**Com.poly.utils**

**PageShow**

1. @Service
2. **public** **class** PageShow {
3. @Autowired
4. ProductDAO prodao;
5. @Autowired
6. ImageDAO imageDAO;
7. @Autowired
8. HttpServletRequest request;
10. **public** **void** pageProductAll(Optional<Integer> p, Optional<String> field, Model model, String pagename) {
11. Sort sort = Sort.by(Direction.ASC, field.orElse("id"));
12. model.addAttribute("field", field.orElse("id").toUpperCase());
13. Pageable pageable;
14. Pageable pageable1;
15. Page<Product> page;
16. **if** (p.orElse(0) < 0) {
17. pageable1 = PageRequest.of(0, 10, sort);
18. page = prodao.findAll(pageable1);
19. request.setAttribute(pagename, page);
20. } **else** {
21. pageable = PageRequest.of(p.orElse(0), 10, sort);
22. page = prodao.findAll(pageable);
24. **if** (p.orElse(0) >= page.getTotalPages()) {
25. pageable1 = PageRequest.of(page.getTotalPages() - 1, 10, sort);
26. } **else** {
27. pageable1 = PageRequest.of(p.orElse(0), 10, sort);
28. }
29. page = prodao.findAll(pageable1);
30. request.setAttribute(pagename, page);
31. }
32. }
34. **public** **void** pageProductSeach(Optional<Integer> p, Optional<String> field, Model model, String pagename,
35. String idpro) {
36. Sort sort = Sort.by(Direction.ASC, field.orElse("id"));
37. model.addAttribute("field", field.orElse("id").toUpperCase());
38. Pageable pageable;
39. Pageable pageable1;
40. Page<Product> page;
41. **if** (p.orElse(0) < 0) {
42. pageable1 = PageRequest.of(0, 10, sort);
43. page = prodao.findByKeywords("%" + idpro + "%", pageable1);
44. request.setAttribute(pagename, page);
45. } **else** {
46. pageable = PageRequest.of(p.orElse(0), 10, sort);
47. page = prodao.findByKeywords("%" + idpro + "%", pageable);
49. **if** (p.orElse(0) >= page.getTotalPages()) {
50. pageable1 = PageRequest.of(page.getTotalPages() - 1, 10, sort);
51. } **else** {
52. pageable1 = PageRequest.of(p.orElse(0), 10, sort);
53. }
54. page = prodao.findByKeywords("%" + idpro + "%", pageable1);
55. request.setAttribute(pagename, page);
56. }
57. }
59. // Image
60. **public** **void** pageImageAll(Optional<Integer> p, Optional<String> field, Model model, String pagename) {
61. Sort sort = Sort.by(Direction.ASC, field.orElse("id"));
62. model.addAttribute("field", field.orElse("id").toUpperCase());
63. Pageable pageable;
64. Pageable pageable1;
65. Page<Image> page;
66. **if** (p.orElse(0) < 0) {
67. pageable1 = PageRequest.of(0, 10, sort);
68. page = imageDAO.findAll(pageable1);
69. request.setAttribute(pagename, page);
70. } **else** {
71. pageable = PageRequest.of(p.orElse(0), 10, sort);
72. page = imageDAO.findAll(pageable);
74. **if** (p.orElse(0) >= page.getTotalPages()) {
75. pageable1 = PageRequest.of(page.getTotalPages() - 1, 10, sort);
76. } **else** {
77. pageable1 = PageRequest.of(p.orElse(0), 10, sort);
78. }
79. page = imageDAO.findAll(pageable1);
80. request.setAttribute(pagename, page);
81. }
82. }
84. **public** **void** pageImageSeach(Optional<Integer> p, Optional<String> field, Model model, String pagename,
85. String idpro) {
86. Sort sort = Sort.by(Direction.ASC, field.orElse("id"));
87. model.addAttribute("field", field.orElse("id").toUpperCase());
88. Pageable pageable;
89. Pageable pageable1;
90. Page<Image> page;
91. **if** (p.orElse(0) < 0) {
92. pageable1 = PageRequest.of(0, 10, sort);
93. page = imageDAO.findByKeywords("%" + idpro + "%", pageable1);
94. request.setAttribute(pagename, page);
95. } **else** {
96. pageable = PageRequest.of(p.orElse(0), 10, sort);
97. page = imageDAO.findByKeywords("%" + idpro + "%", pageable);
99. **if** (p.orElse(0) >= page.getTotalPages()) {
100. pageable1 = PageRequest.of(page.getTotalPages() - 1, 10, sort);
101. } **else** {
102. pageable1 = PageRequest.of(p.orElse(0), 10, sort);
103. }
104. page = imageDAO.findByKeywords("%" + idpro + "%", pageable1);
105. request.setAttribute(pagename, page);
106. }
107. }
109. }

**DataSharing**

1. **public** **class** DataSharing{
3. **public** **static** HashMap<String, Product> product = **new** HashMap<>();
4. **public** **void** addList(List<Product> list) {
5. **for** (**int** i = 0; i < list.size(); i++) {
6. product.put(list.get(i).getId(), list.get(i));
7. }
8. }
10. }

**Com.poly.entity**

**Account**

1. @SuppressWarnings("serial")
2. @Data
3. @Entity
4. @Table(name = "Accounts")
5. @AllArgsConstructor
6. @NoArgsConstructor
7. **public** **class** Account  **implements** Serializable{
8. @Id
9. String username;
10. String password;
11. String fullname;
12. String email;
13. String photo;
14. **boolean** admin;
15. **boolean** activated;
16. @JsonIgnore
17. @OneToMany(mappedBy = "account")
18. List<Ord

**Category**

1. @SuppressWarnings("serial")
2. @Data
3. @Entity
4. @Table(name = "Category")
5. @AllArgsConstructor
6. @NoArgsConstructor
7. **public** **class** Category **implements** Serializable{
8. @Id
9. String id;
10. String name;
11. @JsonIgnore
12. @OneToMany(mappedBy = "category")
13. List<Product> products;
14. @Override
15. **public** String toString() {
16. **return** "Category [id=" + id + ", name=" + name + "]";
17. }

20. }

**Color**

1. @Data
2. @Entity
3. @Table(name = "Color")
4. @AllArgsConstructor
5. @NoArgsConstructor
6. **public** **class** Color **implements** Serializable{
7. @Id
8. @GeneratedValue(strategy = GenerationType.IDENTITY)
9. Integer id;
10. String name;
11. @ManyToOne
12. @JoinColumn(name = "Productid")
13. Product colorprod;
14. @ManyToOne
15. @JoinColumn(name = "Imageid")
16. Image image;
17. @OneToMany(mappedBy = "color")
18. List<OrderDetail> orderDetails;
19. **public** Color(String name, Product colorprod) {
20. **super**();
21. **this**.name = name;
22. **this**.colorprod = colorprod;
23. }
24. }

**Image**

1. @Data
2. @Entity
3. @Table(name = "Image")
4. @AllArgsConstructor
5. @NoArgsConstructor
6. **public** **class** Image **implements** Serializable{
7. @Id
8. String id;
9. String name;
10. @OneToMany(mappedBy = "image")
11. List<Color> colors;
12. }

**Order**

1. @SuppressWarnings("serial")
2. @Data
3. @Entity
4. @Table(name = "Orders")
5. @AllArgsConstructor
6. @NoArgsConstructor
7. **public** **class** Order  **implements** Serializable{
8. @Id
9. @GeneratedValue(strategy = GenerationType.IDENTITY)
10. Long id;
12. @ManyToOne
13. @JoinColumn(name = "Username")
14. Account account;
16. @Temporal(TemporalType.DATE)
17. @Column(name = "Createdate")
18. Date createDate = **new** Date();
20. String address;
21. String phone;
22. Double totalmoney;
24. @JsonIgnore
25. @OneToMany(mappedBy = "order")
26. List<OrderDetail> orderDetails;
27. }

**Brand**

1. @SuppressWarnings("serial")
2. @Data
3. @Entity
4. @Table(name = "Brand")
5. @AllArgsConstructor
6. @NoArgsConstructor
7. **public** **class** Brand **implements** Serializable{
8. @Id
9. String id;
10. String name;
11. @JsonIgnore
12. @OneToMany(mappedBy = "brand")
13. List<Product> products;

**OrderDetail**

1. @SuppressWarnings("serial")
2. @Data
3. @Entity
4. @Table(name = "Orderdetails")
5. @AllArgsConstructor
6. @NoArgsConstructor
7. **public** **class** OrderDetail **implements** Serializable {
8. @Id
9. @GeneratedValue(strategy = GenerationType.IDENTITY)
10. Long id;
12. Double price;
14. Integer quantity;
16. @ManyToOne(cascade = CascadeType.MERGE)
17. @JoinColumn(name = "Orderid", referencedColumnName = "id")
18. @JsonIgnoreProperties(value = { "applications", "hibernateLazyInitializer" })
19. Order order;
21. @ManyToOne(cascade = CascadeType.MERGE)
22. @JoinColumn(name = "Productid", referencedColumnName = "id")
23. @JsonIgnoreProperties(value = { "applications", "hibernateLazyInitializer" })
24. Product oderproduct;
26. @ManyToOne(cascade = CascadeType.MERGE)
27. @JoinColumn(name = "Sizeid", referencedColumnName = "id")
28. @JsonIgnoreProperties(value = { "applications", "hibernateLazyInitializer" })
29. Size size;
31. @ManyToOne(cascade = CascadeType.MERGE)
32. @JoinColumn(name = "Colorid", referencedColumnName = "id")
33. @JsonIgnoreProperties(value = { "applications", "hibernateLazyInitializer" })
34. Color color;
36. }

**Product**

1. @SuppressWarnings("serial")
2. @Data
3. @Entity
4. @Table(name = "Products")
5. @AllArgsConstructor
6. @NoArgsConstructor
7. **public** **class** Product **implements** Serializable{
8. @Id
9. String id;
10. String name;
11. Double price;
12. String info;
13. Boolean available;
14. Integer quantity;
15. @ManyToOne
16. @JoinColumn(name = "Categoryid")
17. Category category;
18. @ManyToOne
19. @JoinColumn(name = "Brandid")
20. Brand brand;
21. @OneToMany(mappedBy = "colorprod",fetch=FetchType.EAGER)
22. List<Color> colors;
23. @OneToMany(mappedBy = "sizeprod")
24. List<Size> sizes;
25. @OneToMany(mappedBy = "oderproduct")
26. List<OrderDetail> orderDetails;
28. }

**Size**

1. @SuppressWarnings("serial")
2. @Data
3. @Entity
4. @Table(name = "Size")
5. @AllArgsConstructor
6. @NoArgsConstructor
7. **public** **class** Size **implements** Serializable{
8. @Id
9. @GeneratedValue(strategy = GenerationType.IDENTITY)
10. Integer id;
11. String name;
13. @ManyToOne
14. @JoinColumn(name = "Productid")
15. Product sizeprod;
17. @OneToMany(mappedBy = "size")
18. List<OrderDetail> orderDetails;
20. **public** Size(String name, Product sizeprod) {
21. **super**();
22. **this**.name = name;
23. **this**.sizeprod = sizeprod;
24. }
25. }