TRƯỜNG ĐẠI HỌC NÔNG LÂM THÀNH PHỐ HỒ CHÍ MINH

Khoa Công Nghệ Thông Tin

🙞🙞🕮🙜🙜

**BÁO CÁO CUỐI KỲ**

MÔN: LẬP TRÌNH FRONT END

**WEBSITE BÁN ĐỒ CHƠI – THẾ GIỚI ĐỒ CHƠI**

Giảng viên: Phan Đình Long

Nhóm 12

Trần Nhật Thy 18130239

Trịnh Quang Tiến 18130243

Dương Bá Nhuận 18130170

Ngày 17 tháng 09 năm 2021

🙚🕮🙘

**BẢNG PHÂN CÔNG**

|  |  |  |  |
| --- | --- | --- | --- |
| Tên | MSSV | Công việc | Tiến độ |
| Trinh Quang Tiến | 18130243 | Trang chủ  Trang Mua sắm  Trang danh sách bài viết  ProductService  PaginationService  BlogService  LoadJsonService,  Breadcrumb,  Crawl dữ liệu: Nhân vật | 100% |
| Dương Bá Nhuận | 18130170 | Header  Trang yêu thích  Trang chi tiết blog  Trang liên hệ  Trang thanh toán  LoveService  Crawl dữ liệu: blog, lego, puzzle. smartgame | 100% |
| Trần Nhật Thy | 18130239 | Trang chi tiết sản phẩm  Footer  Trang giỏ hàng  CartService  CartItem  ScrollToTop  Crawl dữ liệu Đồ chơi vận động và Phương tiện giao thông. | 100% |

MỤC LỤC

[I. GIỚI THIỆU 4](#_Toc82802164)

[1. Mục đích 4](#_Toc82802165)

[2. Phạm vi 4](#_Toc82802166)

[3. Thư viện sử dụng 4](#_Toc82802167)

[4. Nguồn tham khảo 4](#_Toc82802168)

[II. TỔNG QUAN VỀ TRANG WEB 5](#_Toc82802169)

[III. Mô tả kĩ thuật: 8](#_Toc82802170)

[1. Dữ liệu: 8](#_Toc82802171)

[1.1. Lấy dữ liệu: 8](#_Toc82802172)

[1.2. Load file Json từ ứng dụng Angular: 8](#_Toc82802173)

[2. Cách hoạt động của ứng dụng: 8](#_Toc82802174)

[2.1. Các service có trong ứng dụng 8](#_Toc82802175)

[2.2. Nguyên tắc hoạt động theo các trang 10](#_Toc82802176)

[2.3. Cụ thể nguyên tắc hoạt động của các trang: 10](#_Toc82802177)

[2.4. Nguyên tắc hoạt động theo chức năng 16](#_Toc82802178)

[IV. Hướng dẫn sử dụng 18](#_Toc82802179)

[1. Tìm kiếm 18](#_Toc82802180)

[2. Lọc 19](#_Toc82802181)

[2.1. Lọc theo danh mục 19](#_Toc82802182)

[2.2. Lọc theo tuổi hoặc giới tính 19](#_Toc82802183)

[2.3. Lọc theo giá tiền 20](#_Toc82802184)

[3. Sắp xếp: 20](#_Toc82802185)

[4. Thêm vào giỏ hàng 21](#_Toc82802186)

[5. Thêm vào danh sách yêu thích 23](#_Toc82802187)

[6. Thanh toán. 24](#_Toc82802188)

# GIỚI THIỆU

## Mục đích

Giới thiệu về trang web bán đồ chơi có tên là Thế giới đồ chơi.

Mô tả trang web Thế giới đồ chơi và hướng dẫn sử dụng trang web.

## Phạm vi

Đối tượng sử dụng trang web dành cho các phụ huynh có nhu cầu mua đồ chơi cho con, hoặc quà tặng cho các em nhỏ.

## Thư viện sử dụng

## Nguồn tham khảo

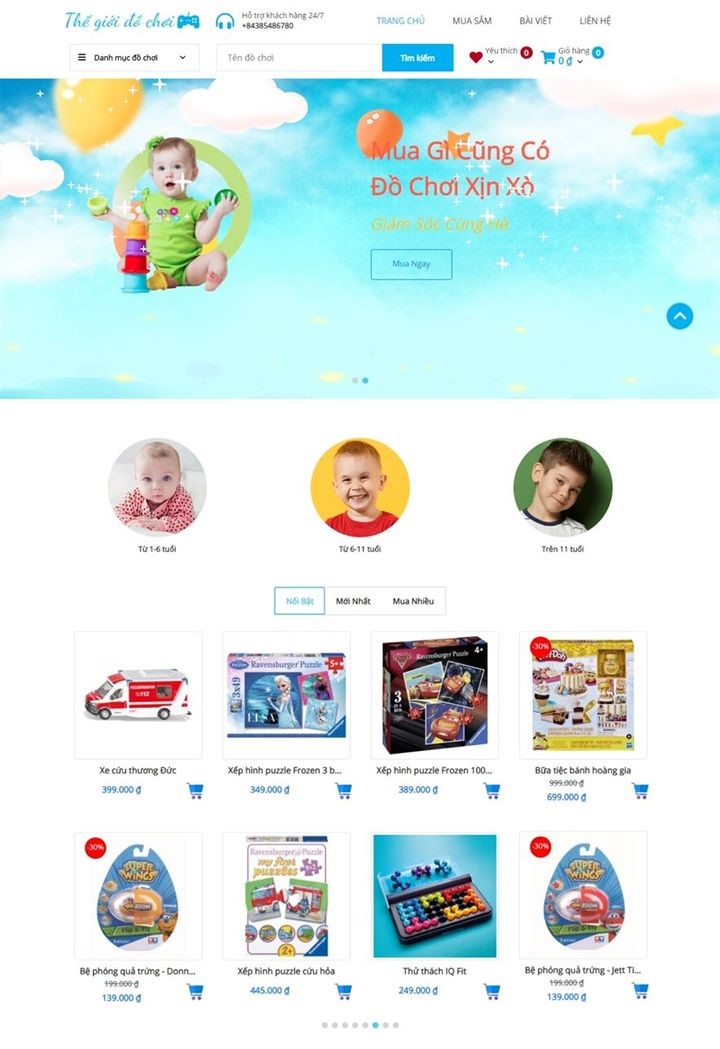
Lấy dữ liệu:

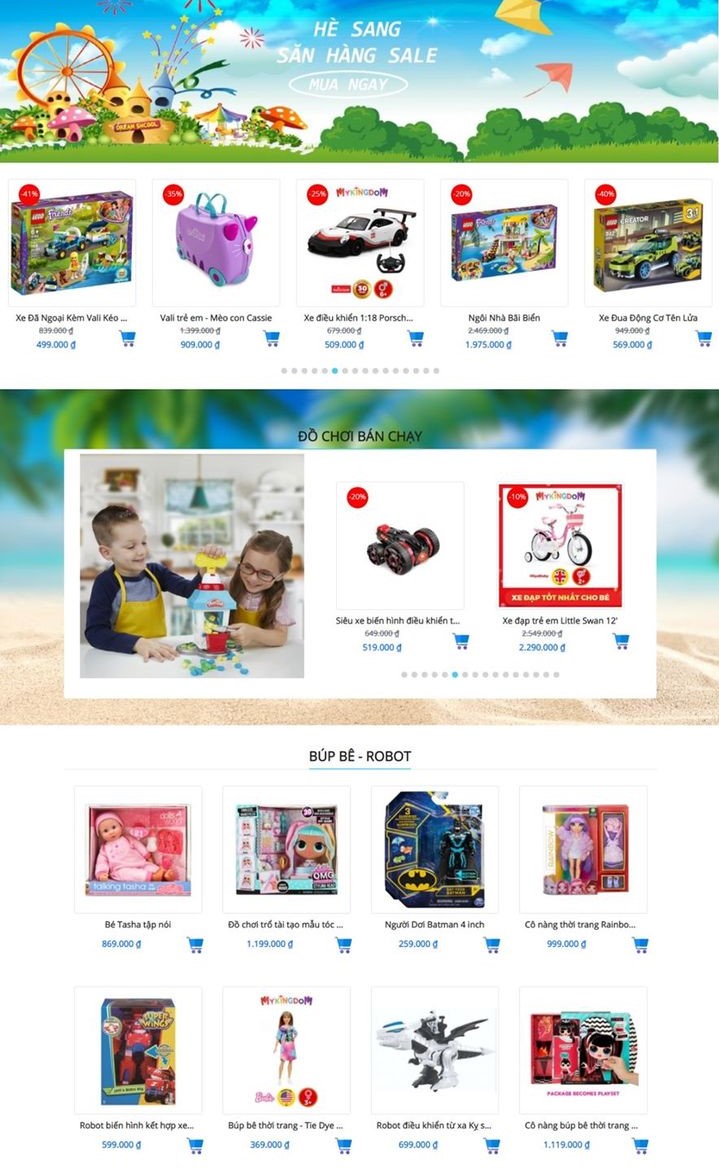
<https://www.mykingdom.com.vn/>

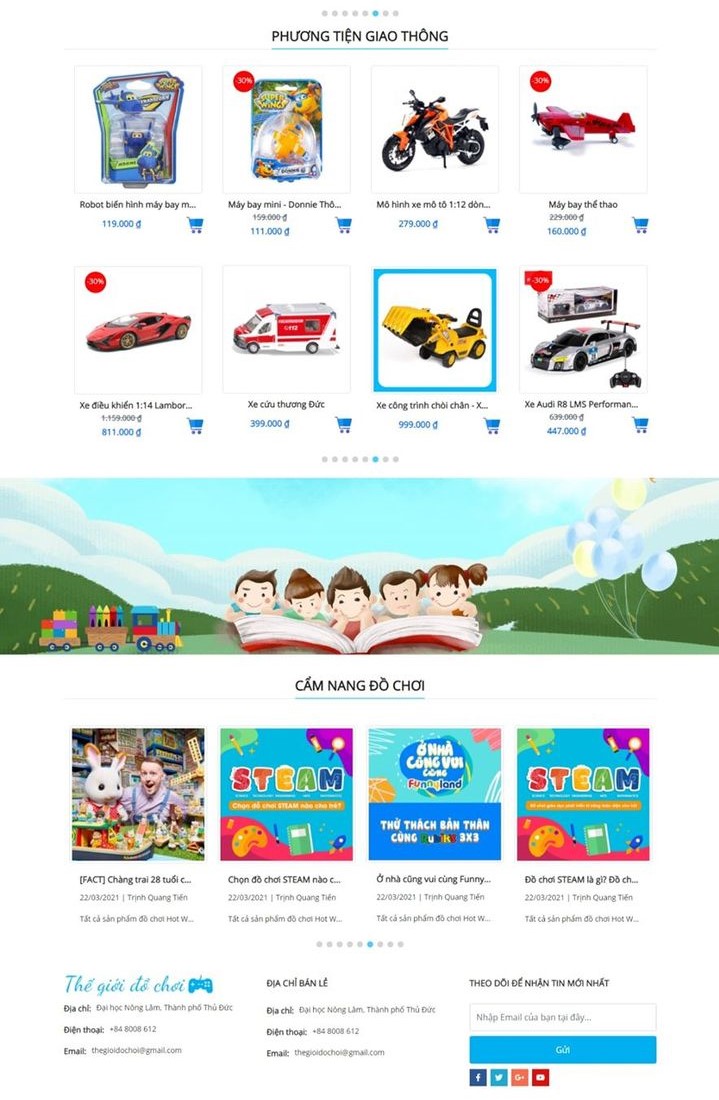
<https://www.funnyland.vn/>

<https://kidsland.vn/>

# TỔNG QUAN VỀ TRANG WEB







# Mô tả kĩ thuật:

## Dữ liệu:

### Lấy dữ liệu:

Mỗi danh mục crawl 20 dữ liệu và lưu thành file Json.

Trang crawl dữ liệu: mykingdom.com.vn, funnyland.vn, kidsland.vn

Phần mềm crawl: Extension của Chrome: Data miner.

Dữ liệu crawl gồm:

* Crawl sản phẩm:

Tên sản phẩm: tên sản phẩm

Giá gốc: giá gốc của sản phẩm

Giá giảm: giá của sản phẩm khi được giảm giá

Thương hiệu: tên thương hiệu của sản phẩm

Mã sản phẩm: mã sản phẩm

Mô tả: mô tả chi tiết về sản phẩm

Độ tuổi: độ tuổi được phân loại theo 3 giới hạn tuổi: từ 1 đến 6 tuổi,  từ 6 đến 11 tuổi, trên 11

Giới tính: sản phẩm đó được chơi đối với giới tính nào

Ảnh: crawl từ 1 đến 4 ảnh của sản phẩm

* Crawl bài viết:

Tiêu đề: Tiêu đề bài viết

Ảnh: ít nhất 1 ảnh của bài viết

Nội dung bài viết

### Load file Json từ ứng dụng Angular:

Sử dụng phương thức get của HttpClient service để load các file Json sau đó phân tích file Json thành đối tượng Product và Blog tương ứng trong ứng dụng.

## Cách hoạt động của ứng dụng:

### Các service có trong ứng dụng

#### ProductService:

Khi người người dùng vừa vào trang web, hệ thống sẽ khởi tạo ProductService và load file search2.json bằng LoadJsonService sau đó gọi phương thức convertObj2ProArr của lớp Util để chuyển đổi danh sách các object sang danh sách các sản phẩm sau đó chờ đăng kí từ các component. ProductService có các phương thức:

Phương thức filter là phương thức lọc các sản phẩm theo điều kiện, phương thức này sẽ được gọi ở ProductComponent để lọc các sản phẩm theo điều kiện, sau khi lọc xong, phương thức sẽ gọi phương thức next của BehaviorSubject để thông báo cho các component đã đăng kí.

Phương thức getRandomProducts là phương thức lấy ngẫu nhiên 4 sản phẩm trong danh sách các sản phẩm.

Phương thức notify là phương thức thông báo cho các component đã đăng kí BehaviorSubject.

Phương thức sortArrIncrease là phương thức sắp xếp danh sách sản phẩm theo giá tăng dần.

Phương thức sortArrDecrease là phương thức sắp xếp danh sách sản phẩm theo giá giảm dần.

Phương thức findProductById là phương thức tìm kiếm sản phẩm dựa vào productId.

#### BlogService:

Khi người người dùng vừa vào trang web, hệ thống sẽ khởi tạo BlogService và load file blog.json bằng LoadJsonService sau đó gọi phương thức convertObj2BlogArr của lớp Util để chuyển đổi danh sách các object sang danh sách các bài viết sau đó chờ đăng kí từ các component. **BlogService** có các phương thức:

Phương thức getAllBlogs: là phương thức thiết lập chỉ số trang hiện tại của danh sách blog và kích hoạt phương thức notify để lấy danh sách các bài viết.

Phương thức notify là phương thức thông báo cho các component đã đăng kí BehaviorSubject.

Phương thức findBlogById là phương thức tìm kiếm danh sách bài viết dựa vào blogId

#### LoadJsonService

Là service dùng để load file json bằng HttpClient. Trong service có phương thức getData để load file json.

PaginationService: là service dùng để phân trang sản phẩm và phân trang bài viết. Trong service này có phương thức setUpPagination để thiết lập phân trang và phương thức pagination để lấy đối tượng pagination.

#### LoveService

Khi người dùng vừa vào trang web, hệ thống sẽ khởi tạo LoveService. LoveService có các phương thức: addLove, deleteItemLove.

Phương thức addLove dùng để thêm mới sản phẩm yêu thích vào danh sách sản phẩm yêu thích

Phương thức deleteItemLove dùng để xóa một sản phẩm khỏi danh sách sản phẩm yêu thích.

#### CartService

Khi người dùng vừa vào trang web, hệ thống sẽ khởi tạo CartService, trong đó có các phương thức là addCart, deleteItemProduct, resetCart.

Phương thức addCart dùng để chuyển một sản phẩm thành CartItem và lưu vào cart (danh sách CartItem), nếu sản phẩm chưa tồn tại sẽ thêm vào, còn nếu đã tồn tại thì thay đổi số lượng.

Phương thức deleteItemProduct dùng để xóa CartItem ở vị trí đó ra khỏi cart.

Phương thức resetCart dùng để xóa toàn bộ CartItem trong cart.

### Nguyên tắc hoạt động theo các trang

Tổng thể ứng dụng được chia thành 6 module và được load theo phương thức lazy load, 6 module gồm:

App module: gồm Trang chủ, trang Not Found, thanh Header, thanh Footer

ProductPage module: gồm trang Sản phẩm, trang Chi tiết sản phẩm

BlogPage module: gồm trang Danh sách bài viết, trang Chi tiết bài viết

Contact moudle: gồm trang Liên hệ

Cart module: gồm trang Giỏ hàng và trang Thanh toán

Love module: gồm trang Yêu thích

### Cụ thể nguyên tắc hoạt động của các trang:

#### Thanh Header:

Trang header thuộc AppModule, bao gồm một component header

Thanh header sẽ đăng ký các service của CartService, LoveService cập nhật những thay đổi trong danh sách sản phẩm ở giỏ hàng hay danh sách sản phẩm yêu thích để hiển thị những danh sách sản phẩm mới thêm vào, hay khi thêm mới một sản phẩm yêu thích.

Ngoài ra sẽ tính toán số lượng sản phẩm yêu thích đã thêm vào bằng cách gọi phương thức numberOfListProduct, số lượng sản phẩm hiện có trong giỏ hàng bằng cách gọi phương thức numberOfListLoveProduct từ đó tính ra tổng giá tiền cũng như cập nhật những thay đổi từ phía người dùng bằng phương thức totalPriceOfAllProduct.

Mỗi hình ảnh xuất hiện của mỗi sản phẩm ở nơi hiển thị danh sách các sản phẩm yêu thích hay sản phẩm ở giỏ hàng thì sẽ gọi phương thức getFirstImage để chỉ lấy hình ảnh đầu tiên của mỗi sản phẩm.

Xem danh sách sản phẩm ở giỏ hàng bằng di chuyển chuột vào vị trí giỏ hàng, lúc này trang web sẽ nhận sự kiện hover, sau đó danh sách sản phẩm sẽ hiện ra.

Xem danh sách sản phẩm yêu thích cũng tương tự cách xem danh sách sản phẩm, trang web sẽ nhận sự kiện hover khi di chuột vào mục yêu thích và danh sách yêu mà người dùng thêm vào sẽ hiện ra.

Thanh header highlight bằng cách nhận đường link hiện tại từ router, sau đó sẽ tách ra để tìm tên trang hiển thị hiện tại ví dụ: home, contact, blog, contact. Tại trang header sẽ highlight tên trang đang hiển thị trên trang web bằng class.active tương ứng với trang đang hiển thị.

Thanh header sẽ routing đến các trang khác ở trong trang web bằng cách khai báo routing module cho mỗi trang web khác nhau, để khi nhấn vào trang web sẽ routing chính xác đến trang đó.

Tại mục tìm kiếm của header, Khi người dùng gõ tên của một tên sản phẩm bất kì sau đó nhấn enter hoặc nút tìm kiếm thì chương trình sẻ gọi phương thức searchRoute để lấy chuỗi kí tự mà người dùng vừa nhập vào, tiếp theo theo đó sẻ khởi tạo một link để tìm kiểm bằng phường thức makeLinkProduct. Và cuối cùng gọi router routing bằng đường link vừa tạo để tìm xem có sản phẩm nào mà trong tên có kí tự trùng với từ khóa người dùng nhập vào hay không.

Hiển thị tùy chọn danh mục đồ chơi bằng cách click chuột vào danh mục đồ chơi, khi đó trang web sẽ nhận sự kiện click, và khởi chạy phương thức showslide để hiện thị tên các danh mục. Sau đó di chuyển chuột vào từng tên danh mục, trang web sẽ nhận sự kiện hover và hiện thị tên từng danh mục con.

#### Trang liên hệ

Trang liên hệ thuộc AppModule, trang có 2 component đó là:

ContactForm Component là component dùng để người dùng nhập thông tin liên hệ bao gồm: Họ và tên, email, số điện thoại và ghi chú

ContactAddressComponent là component dùng để hiển thị thông tin của chủ sử hữu trang web bao gồm Địa chỉ, email, số điện thoại

#### Trang thanh toán

Trang thanh toán thuộc CartCheckout module, bao gồm một component là:

Checkout component bao gồm địa chỉ nhận hàng và hóa đơn của khách hàng.

Địa chỉ nhận hàng bao gồm lưu trữ các thông tin: Họ, tên, email, địa chỉ, ghi chú.

Hóa đơn khách hàng sẽ hiển thị danh sách sản phẩm đặt hàng bao gồm các thông tin tên sản phẩm, số lượng, đơn giá và tổng giá tiền sản phẩm sau khi cộng tiền ship.

Trang thanh toán sẽ đăng kí CartService để lấy thông tin danh sách sản phẩm đang có trong giỏ hàng. Tổng tiền của giỏ hàng sẽ tính bằng cách gọi phương thức totalPriceOfAllProduct.

#### Trang chi tiết bài viết

Trang chi tiết bài viết thuộc BlogPageModule, bao gồm một component là BlogDetailComponent.

Thang chi tiết bài viết hiển thị thông tin bao gồm Tiêu đề của bài viết, Nội dung của bài viết

Trang chi tiết bài viết lấy thông tin bài viết bằng cách gọi phương thức findBlogById của blogService để lấy nội dung của blog.

#### Trang yêu thích

Trang yêu thích thuộc AppModule, bao gồm một LoveComponent và một LoveableComponent

LoveComponent là component dùng để hiển thị danh sách các sản phẩm yêu thích của người dùng: Thông tin sản phẩm bao gồm: hỉnh ảnh sản phẩm, tên, giá.

LoveComponent lấy thông tin sản phẩm yêu thích bằng cách đăng kí đăng kí với LoveService để lấy danh sách sản phẩm yêu thích của khách hàng. Chức năng xóa một sản phẩm yêu thích của khách hàng bằng cách gọi deleteLoveItem để xóa sản phẩm đó.

LoveableComponent là component dùng để hiện thị khoảng 4 sản phẩm bất kì để gợi ý cho người dùng.

LoveableComponent lấy sản phẩm này bằng cách gọi phương thức bằng cách đăng kí ProductService rồi gọi phương thức getRandomProduct.

#### Trang chi tiết sản phẩm

Trang chi tiết sản phẩm thuộc ProductModule, trang có các component:

ProductCardDetailComponent: là component hiển thị hình ảnh và chi tiết sản phẩm.

DescriptionComponent: là component hiển thị mô tả sản phẩm.

RelatedProductComponent: là component hiển thị các sản phẩm tương tự.

Khi nhấn vào sản phẩm ở một trang danh sách sản phẩm, trang sẽ router đến link có product\_id của sản phẩm đó. Tại trang chi tiết sản phẩm có input là một product, khi router đến sẽ load lên thông tin của sản phẩm có product\_id đó. Trang có hai button là Thêm vào giỏ hàng và Thêm vào yêu thích, tại đây trang có gọi CartService và LoveService để thực hiện handleAddToCart và OnClickLoveProduct. HandleAddToCart sẽ gọi phương thức addCart của CartService để thêm sản phẩm đó và số lượng vào danh sách giỏ hàng của CartService. OnClickLove Product sẽ gọi phương thức addLove để thêm sản phẩm đó vào danh sách yêu thích của LoveService.

Tương tự phần Mô tả sản phẩm cũng load lên thông tin mô tả của sản phẩm có product\_id đó.

Phần sản phẩm tương tự sẽ gọi getRandomProduct của ProductService để lấy ngẫu nhiên 4 sản phẩm từ json.

#### Trang giỏ hàng

Trang giỏ hàng thuộc CartCheckoutModule, trang có các component:

CartTableComponent: là component hiển thị danh sách các sản phẩm đã thêm vào giỏ hàng.

CartTotalComponent: là component hiển thị bảng tóm tắt giá tiền của giỏ hàng.

ProductLikeComponent: là component hiển thị một số sản phẩm.

Trang giỏ hàng, phần CartTable và CartTotal subscribe Observable cart$ của CartService để lấy thông tin mà các sản phẩm đã thêm vào, các sản phẩm đã thêm vào sẽ được lưu vào một danh sách ở CartService bằng addCart thành từng CartItem. CartItem gồm các thuộc tính: name, image, price, quantity, product\_id. Nếu sản phẩm chưa có ở giỏ hàng, sẽ được thêm vào, còn nếu sản phẩm đã có ở giỏ hàng thì sẽ được cộng thêm số lượng.

CartTable có một danh sách CartItem bằng với danh sách CartItem của CartService, CartTable subscribe Observable cart$ của CartService để có danh sách khi có sự thay đổi ở trang khác. CartTable có nút tăng, giảm số lượng sản phẩm, nếu giảm số lượng sản phẩm < 0 sẽ xóa sản phẩm, lúc này gọi deleteItemProduct của CartService để xóa khỏi danh sách CartItem của CartService. Ở mỗi sản phẩm cũng có button xóa từng sản phẩm, gọi deleteItemProduct của CartService để xóa.

CartTotal gọi totalPrice của CartService để tính tổng tiền của các sản phẩm trong giỏ hàng (chưa có phí vận chuyển).

ProductLike gọi getRandomProduct của ProductService để lấy ngẫu nhiên 2 sản phẩm.

#### Thanh footer

Thanh footer thuộc AppModule, trang có component FooterComponent.

#### Trang chủ

Trang chủ thuộc HomeModule, trang có các component:

HomeComponent: là component cha của các component sau:

HeroAreaComponent: là component hiển thị banner ở đầu trang.

AgeAreaComponen: là component hiển thị các banner độ tuổi, người dùng có thể bấm vào hình ảnh banner để route đến trang sản phẩm.

FeatureHomeComponent: là component hiển thị các sản phẩm Nổi bật, sản phẩm Mới nhất, Mua nhiều. Component này có các input: best saller là danh sách các sản phẩm bán chạy; latest là danh sách các sản phẩm mới nhất; hightlight: là danh sách các sản phẩm nổi bật. Component này có output: getProductQuickViewOutput: output này giúp truyền sản phẩm được click trong phần xem nhanh chi tiết sản phẩm ra HomeComponent để hiển thị hộp modal xem nhanh của sản phẩm được click.

BestSallerComponent: là component hiển thị các sản phẩm được mua nhiều nhất. Component này có input: bestSaller là danh sách các sản phẩm bán chạy. Component này có output là getProductQuickViewOuput, output này có chức năng tương tự như ở FuildBannerComponent là component hiển thị hình ảnh banner. Component này nhận vào input: img\_src: là đường dẫn đến ảnh hiển thị; url: là đường dẫn để route

PersonTrafficComponent là component hiển thị danh sách các sản phẩm thuộc danh mục Nhân vật và danh mục Phương tiện giao thông. Component này nhận vào các input: sectionTitle: Tiêu đề của section; products: danh sách cách sản phẩm hiển thị. Component này có output: getProductQuickViewOuput, output này giống với ouput ở phần FeatureHomeComponent

SaleAreaComponent: là component hiển thị danh sách các sản phẩm giảm giá. Component này có input: saleProducts: là danh sách các sản phẩm giảm giá. Component này có output: getProductQuickViewOutput.

BlogAreaComponent là component hiển thị danh sách các bài viết. Component này có input: blogs là danh sách các bài viết.

Khi vừa vào trang chủ hệ thống sẽ load danh sách các sản phẩm ở file home.json sau đó dùng phương thức convertObj2ProArr của lớp Util để chuyển đổi đối tượng danh sách các đối tượng object thành danh sách các Product và Blog.

Product là class có các thuộc tính: name, price, price\_sale, brand, product\_id, description, age, gender, images, category.

Blog là interface có các thuộc tính: name, date, author, content, img, id.

Sau khi load xong file home.json, hệ thống sẽ thêm các input vào trang html để truyền đến các component con.

Ở trang chủ sử dụng thư viện ngx-slick-carousel tạo chuyển động băng chuyền cho các item sản phẩm.

Mỗi item sản phẩm gồm hình ảnh của sản phẩm, giá bán sản phẩm, giá giảm (nếu có), biểu tượng giỏ hàng, biểu tượng thêm vào danh sách yêu thích, biểu tượng xem nhanh chi tiết sản phẩm.

Khi người dùng bấm vào nút thêm giỏ hàng ở từng item sản phẩm, hệ thống sẽ gọi cart service để thêm sản phẩm vào giỏ hàng, đồng thời hiện hiển thị hiệu ứng đang thêm vào giỏ hàng trong vòng 1 giây.

Khi người di chuyển chuột vào vùng item sản phẩm sẽ có biểu tượng hình con mắt hiển thị trên ảnh của item sản phẩm đó, người dùng bấm vào biểu tượng hình con mắt để xem nhanh chi tiết sản phẩm đó. Ở modal xem nhanh chi tiết sản phẩm người dùng có bấm vào nút yêu thích, hoặc nút thêm vào giỏ hàng.

Khi người dùng di chuyển chuột vào vùng item sản phẩm sẽ có biểu tượng hình trái tim hiển thị trên ảnh của item sản phẩm đó, người dùng bấm vào biểu tượng hình trái tim để thêm sản phẩm vào danh sách sản phẩm yêu thích. Khi bấm vào nút yêu thích hệ thống sẽ gọi love service để thêm sản phẩm vào danh sách yêu thích.

Khi người dùng di chuyển chuột vào vùng hình ảnh của item sản phẩm đang hiển thị sẽ bị mất thay và được thay bằng hình ảnh khác. Khi người dùng di chuyển chuột ra khỏi vùng hình ảnh đó, hình ảnh cũ sẽ hiển thị lại.

Người dùng bấm vào hình ảnh hoặc tiêu đề sản phẩm để route đến trang chi tiết sản phẩm

#### Trang mua sắm

Trang danh sách sản phẩm thuộc module ProductPageModule. Trong đó có ProductComponent là component lấy dữ liệu của activateRoute để truyền cho các component con. Khi vừa load vào component, component sẽ chờ 500 mili giây sau đó dùng queryparam của activateRoute để lấy các param. Các param gồm: category, startPrice, endPrice, search, sort, page. Các param này chính là điều kiện đã được người dùng chọn để lọc trước đó, sau khi lấy được các param hệ thống sẽ gọi phương thức filter của ProductService và bảo vào phương thức các param đó để thực hiện việc lọc. Sau khi ProductService lọc xong sẽ thông báo cho ProductListComponent để hiển thị danh sách sản phẩm đã lọc được. Đồng thời component cũng truyền các param vào phần input của các component con của component này. Các component con gồm:

AgeGenderComponent: là component lọc các sản phẩm theo độ tuổi hoặc giới tính.

Component này có các input: title: là tiêu đề của component; content là mảng các mục để lọc, age là danh sách các độ tuổi người dùng đã yêu cầu lọc, gender là danh sách các giới tính người dùng đã yêu cầu lọc, category là danh mục người dùng đã yêu cầu lọc, startPrice là giá nhỏ nhất khi lọc, endPrice là giá lớn nhất khi lọc, search là chuỗi tìm kiếm người dùng đã tìm kiếm, sort: là kiểu sắp xếp các sản phẩm. Các input trên giúp giữ nguyên những điều kiện lọc trước đó của người dùng. Khi người dùng bấm vào ô lọc nào đó, hệ thống sẽ gọi phương thức makeLinkProduct của lớp Util và truyền vào các điều kiện lọc để lấy được link để route, sau đó gọi phương thức navigateByUrl và bỏ vào link vừa tạo được.

CategoriesComponent: là component lọc các sản phẩm theo danh mục sản phẩm. Component này có các input giống AgeGenderComponent giữ nguyên điều kiện lọc trước đó.  Khi người dùng bấm vào danh mục để lọc, hệ thống sẽ gọi phương thức makeLinkProduct của lớp Util và truyền vào các điều kiện lọc để lấy được link để route, sau đó gọi phương thức navigateByUrl và bỏ vào link vừa tạo được.

ProductListComponent: đây là component hiển thị các sản phẩm đã được lọc. Component này sẽ đăng kí product$ observable của ProductService để nhận các sản phẩm sau khi thỏa điều kiện lọc của ProductService. Component này cũng đăng ký pagination$ observable của PaginationService để nhận các số được phân trang. Sau khi nhận được thông báo từ các nguồn đã đăng kí như trên, component sẽ hiển thị sản phẩm theo đúng điều kiện lọc.

ProductRangeSlideComponent: component này lọc các sản phẩm theo giá, người dùng có thể chọn giá thấp nhất và cao nhất  để lọc các sản phẩm. Component này có các input giống AgeGenderComponent giữ nguyên điều kiện lọc trước đó. Khi người dùng chọn giá để lọc, hệ thống sẽ gọi phương thức makeLinkProduct của lớp Util và truyền vào các điều kiện lọc để lấy được link để route, sau đó gọi phương thức navigateByUrl và bỏ vào link vừa tạo được.

ShopToolbarComponent: là component hiển thị số sản phẩm trong 1 trang, số trang hiển thị sản phẩm và sắp xếp sản phẩm theo giá tăng dần hoặc giảm dần. Component này có các input giống AgeGenderComponent giữ nguyên điều kiện lọc trước đó. Khi người dùng chọn các chế độ sắp xếp, hệ thống sẽ gọi phương thức makeLinkProduct của lớp Util và truyền vào các điều kiện lọc để lấy được link để route, sau đó gọi phương thức navigateByUrl và bỏ vào link vừa tạo được.

Người dùng bấm vào item sản phẩm bất kì để xem thông tin chi tiết của sản phẩm.

#### Trang danh sách bài viết

Trang danh sách bài viết thuộc module BlogPageModule. Trong đó có BlogComponent là component lấy dữ liệu của activateRoute để lấy param số trang hiện tại. Khi vừa load vào component, component sẽ chờ 500 mili giây sau đó dùng queryparam của activateRoute để lấy param page - chỉ trang hiện tại của phân trang. Sau khi lấy được chỉ số trang hiện tại, hệ thống gọi phương thức getAllBlogs của BlogService và truyền vào chỉ số trang hiện tại để thiết lập phân trang và gọi phương thức next của behaviorSubject của BlogService để truyền danh sách bài viết đến chính component này. Sau đó component đăng kí blog$ observable để nhận danh sách các bài viết của blog service. Đồng thời component cũng đăng kí pagination$ observable để đăng kí nhận phân trang.

### Nguyên tắc hoạt động theo chức năng

#### Tìm kiếm

Tại mục tìm kiếm của header, Khi người dùng gõ tên của một tên sản phẩm bất kì sau đó nhấn enter hoặc nút tìm kiếm thì chương trình sẻ gọi phương thức searchRoute để lấy chuỗi kí tự mà người dùng vừa nhập vào, tiếp theo theo đó sẽ khởi tạo một link để tìm kiểm bằng phường thức makeLinkProduct. Sau đó trang điều hướng đến trang Mua sắm, khi điều hướng đến trang mua sắm hệ thống sẽ vào ProductComponent khi đó hệ thống sẽ query các param để lấy chuỗi tìm kiếm. Sau đó gọi phương thức filter của ProductService để lọc theo điều kiện. Sau khi lọc xong ProductService sẽ truyền danh sách các sản phẩm thỏa điều kiện đến các component đã đăng kí trước đó.

#### Lọc

Lọc theo danh mục: Khi người dùng trang Mua sắm và bấm vào một danh mục nào đó thì chương trình sẽ gọi phương thức filterByCategory để lấy tên danh mục người dùng muốn lọc, tiếp theo theo đó sẽ khởi tạo một link để tìm kiểm bằng phường thức makeLinkProduct. Sau đó trang điều hướng đến trang Mua sắm, khi điều hướng đến trang mua sắm hệ thống sẽ vào ProductComponent khi đó hệ thống sẽ query các param để lấy các param làm điều kiện lọc. Sau đó gọi phương thức filter của ProductService để lọc theo điều kiện. Sau khi lọc xong ProductService sẽ truyền danh sách các sản phẩm thỏa điều kiện đến các component đã đăng kí trước đó.

Lọc theo độ tuổi và giới tính: Khi người dùng tích chọn vào độ tuổi hoặc giới tính nào đó, chương trình sẽ gọi phương thức check để lấy tên độ tuổi hoặc giới tính người dùng muốn lọc, tiếp theo theo đó sẽ khởi tạo một link để tìm kiểm bằng phường thức makeLinkProduct. Sau đó trang điều hướng đến trang Mua sắm, khi điều hướng đến trang mua sắm hệ thống sẽ vào ProductComponent khi đó hệ thống sẽ query các param để lấy các param làm điều kiện lọc. Sau đó gọi phương thức filter của ProductService để lọc theo điều kiện. Sau khi lọc xong ProductService sẽ truyền danh sách các sản phẩm thỏa điều kiện đến các component đã đăng kí trước đó.

Lọc theo giá tiền: Khi người dùng chọn vào khoảng giá tiền mong muốn, chương trình sẽ gọi phương thức getValue để lấy tên khoảng giá tiền người dùng muốn lọc, tiếp theo theo đó sẽ khởi tạo một link để tìm kiểm bằng phường thức makeLinkProduct. Sau đó trang điều hướng đến trang Mua sắm, khi điều hướng đến trang mua sắm hệ thống sẽ vào ProductComponent khi đó hệ thống sẽ query các param để lấy các param làm điều kiện lọc. Sau đó gọi phương thức filter của ProductService để lọc theo điều kiện. Sau khi lọc xong ProductService sẽ truyền danh sách các sản phẩm thỏa điều kiện đến các component đã đăng kí trước đó.

Sắp xếp: Khi người dùng chọn sắp xếp sản phẩm tăng dần hoặc giảm dần, chương trình sẽ gọi phương thức sortProduct để lấy điều kiện sắp xếp, tiếp theo theo đó sẽ khởi tạo một link để tìm kiểm bằng phường thức makeLinkProduct. Sau đó trang điều hướng đến trang Mua sắm, khi điều hướng đến trang mua sắm hệ thống sẽ vào ProductComponent khi đó hệ thống sẽ query các param để lấy các param làm điều kiện sắp xếp. Sau đó gọi phương thức filter của ProductService để sắp xếp theo điều kiện. Sau khi sắp xếp xong ProductService sẽ truyền danh sách các sản phẩm thỏa điều kiện đến các component đã đăng kí trước đó.

#### Thêm vào giỏ hàng

Khi nhấn vào biểu tượng giỏ hàng của một sản phẩm bất kỳ trong trang web, tại biểu tượng có phương thức handleAddToCart gọi phương thức addCart của CartService để lưu sản phẩm đó vào danh sách CartItem của CartService. Khi vào giỏ hàng, các sản phẩm có trong danh sách CartItem của CartService sẽ load lên. CartTable và CartTotal subscribe Observable cart$ của CartService để lấy được danh sách CartItem khi có cập nhật, thay đổi trong giỏ hàng. CartTotal sẽ gọi phương thức totalProduct để tính giá tiền của tất cả sản phẩm trong giỏ hàng (chưa có phí vận chuyển).

Hoặc khi nhấn vào button Thêm vào giỏ hàng và thay đổi số lượng của Trang chi tiết sản phẩm, tại button Thêm vào giỏ hàng có phương thức handleAddToCart gọi phương thức addCart của CartService để thêm sản phẩm đó vào danh sách CartItem của CartService.

#### Thêm vào danh sách yêu thích

Khi nhấn vào biểu tượng hình trái tim trên một sản phẩm để thêm sản phẩm đó vào danh sách sản phẩm yêu thích, hệ thống sẻ gọi phương thức addLove ở trong LoveService để thêm sản phẩm mới vào danh sách yêu thích.

Thanh toán: Hóa đơn khách hàng sẽ hiển thị danh sách sản phẩm đặt hàng bao gồm các thông tin tên sản phẩm, số lượng, đơn giá và tổng giá tiền sản phẩm sau khi cộng tiền ship.

Trang thanh toán sẽ đăng kí CartService để lấy thông tin danh sách sản phẩm đang có trong giỏ hàng. Tổng tiền của giỏ hàng sẽ tính bằng cách gọi phương thức totalPriceOfAllProduct.

# Hướng dẫn sử dụng

## Tìm kiếm

Bước 1: Người dùng nhập thông tin tên sản phẩm cần tìm kiêm vào ô search box

Graphical user interface

Description automatically generated with low confidence

Bước 2: Người dùng nhấn enter hoặc nhấn vào nút tìm kiếm để bắt đầu quá trình tìm kiếm

A picture containing application

Description automatically generated

Bước 3: Người dùng xem kết quả tìm kiếm sau khi quá trình tìm kiếm hoàn tất

Graphical user interface, application

Description automatically generated

## Lọc

### Lọc theo danh mục

Bước 1: Tại trang Mua sắm, người dùng bấm vào danh mục bất kì muốn lọc:

Graphical user interface, application, website

Description automatically generated

Sau đó hệ thống sẽ hiển thị kết quả sau khi lọc

### Lọc theo tuổi hoặc giới tính

Bước 1: Tại trang Mua sắm người dùng tích vào độ tuổi hoặc giới tính muốn lọc

Graphical user interface, application, website

Description automatically generated

Sau đó hệ thống sẽ hiển thị kết quả sau khi lọc

### Lọc theo giá tiền

Bước 1: Tại trang Mua sắm, người dùng chọn khoảng giá tiền muốn lọc

Graphical user interface, application, website

Description automatically generated

Sau đó hệ thống sẽ hiển thị kết quả sau khi lọc

## Sắp xếp:

Bước 1: Tại trang Mua sắm, người dùng chọn vào điều kiện sắp xếp

Graphical user interface, application, website

Description automatically generated

Sau đó hệ thống sẽ hiển thị kết quả sau khi lọc

## Thêm vào giỏ hàng

Bước 1: Người dùng nhấn vào nút  để thêm sản phẩm đó vào giỏ hàng.

Graphical user interface, text

Description automatically generatedGraphical user interface, application

Description automatically generated

Hoặc người dùng nhấn vào nút ở trang Chi tiết sản phẩm để thêm sản phẩm đó vào giỏ hàng.

Graphical user interface, application

Description automatically generated

Bước 2: Người dùng nhấn vào Giỏ hàng ở phần đầu trang để đi đến trang Giỏ hàng.

Chart

Description automatically generated

Rồi chọn Giỏ hàng.

Graphical user interface, application

Description automatically generated

Bước 3: Người dùng có thể tăng, giảm số lượng ở nút (1) và xóa sản phẩm trong giỏ hàng ở nút (2).

A screenshot of a computer

Description automatically generated with medium confidence

## Thêm vào danh sách yêu thích

Bước 1: Người dùng nhấn chuột vào biểu tượng hình trái tim  trên mỗi sản phẩm.

Graphical user interface, application

Description automatically generated

Hoặc người dùng có thể nhấn vào button thêm vào yêu thích A picture containing text

Description automatically generated

ở trang chi tiết sản phẩm

Graphical user interface, text, website

Description automatically generated

Bước 2: Người dùng đến trang danh sách sản phẩm yêu thích bằng cách cách nhấn vào button yêu thích khi di chuyển chuột vào mục yêu thích ở header

Graphical user interface, text, application, website

Description automatically generated

Bước 3: Người dùng có thể xóa sản phẩm yêu thích ở trang yêu thích bằng cách nhấn chuột vào biểu tượng xóa ở mỗi sản phẩm.

Table

Description automatically generated

## Thanh toán.

Bước 1: Người dùng chọn vào nút thanh toán ở trong header

Graphical user interface

Description automatically generated with medium confidence

Hoặc người dùng có thể nhấn chọn thanh toán ở trang danh sách giỏ hàng

Graphical user interface, application, Word

Description automatically generated

Bước 2: Người dùng nhập thông tin địa chỉ và kiểm tra lại danh sách đơn hàng

Graphical user interface, text, application, email, website

Description automatically generated

Bước 3: Nhấn vào nút thanh toán ngay và màn hình trả về kết quả thanh toán