**FUNCTIONS IN MVC**

fashe.herokuapp.com

# Tổng quan

Dùng kiến trúc MVC, source code được chia thành 3 phần riêng biệt:

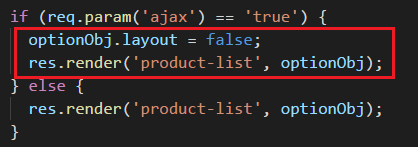
* Model: Thao tác trên CSDL, các hàm thông dụng cho việc truy vấn CSDL. Kết quả trả về thường là Object hoặc mảng các Object có cấu trúc (danh sách thuộc tính) giống với định nghĩa CSDL. Tham số nhận vào thường là một Object truy vấn gần giống với object truy vấn dùng trong các hàm update(), find() của mongoose. Cần xử lý lại các thuộc tính của object này để có thể sử dụng cho truy vấn CSDL.
* View: Các hàm xử lý việc render trang HTML tùy vào các tham số của request (GET, POST) từ trình duyệt.
* Controller:
  + business.js: Xử lý các yêu cầu của một trang web bán hàng. Thiên về xử lý nghiệp vụ, quy trình. Các hàm ở đây trả về mảng các Object như định nghĩa trong Schema. Tham số đầu vào rời rạc (String, Number, Date,… Không có kiểu Object) vì được gọi từ các middleware trong “./routes”
  + “./routes”: Routing cho trang web. Dùng các tham số phương thức GET hoặc POST để gọi hàm render thích hợp.

# View

*Không dùng các hàm ở Model*

## Mô tả

* Dùng module “express-handlebars”.
* Các helper được cài đặt trong file “.\views\helpers\helpers.js”.
* Render cho request AJAX thì set thuộc tính “layout” của object option là *false.*



* Đổi nội dung trang web khi nhận response từ AJAX xử lý tại client. Hàm xử lý cài đặt trong file “.\public\js\client.js”

## client.js

“.\public\js\client.js”

Các hàm xử lý phân trang, kiểm tra dữ liệu hợp lệ (lúc đăng ký tài khoản, điền thông tin nhận hàng…).

Các hàm này được xử lý trên trình duyệt của Client

## Trang HTML

### index.html (Trang chủ)

#### Nội dung hiển thị

Hiển thị những sản phẩm bán chạy nhất.

Giới thiệu website

Thông tin trang trí, quảng cáo…

#### Tham số trong Request GET

### product.html (Trang chi tiết 1 sản phẩm)

#### Nội dung hiển thị

Hiển thị hình ảnh, tên, mô tả, giá, màu, size

Các sản phẩm liên quan (related products)

Nút “Thêm vào giỏ”

#### Tham số trong Request GET

* product: \_id của Product

vd: product=5b23d24993cabc1ae42c46f3

### product-list.html (Trang danh sách sản phẩm/kết quả tìm kiếm/lọc theo loại sản phẩm)

#### Nội dung hiển thị

Hiển thị danh sách sản phẩm (Toàn bộ sản phẩm hoặc đã lọc theo loại, giá, size, màu…)

Ô lọc sản phẩm (Chọn mức giá, màu, size, category…)

Pagination (phân trang kết quả. Số sản phẩm hiển thị 1 trang quy định trong “./config.js” – hằng *PRODUCT\_PER\_PAGE* (Dùng AJAX)

#### Tham số trong Request GET

* category: \_id của Category

vd: category=5b23d24993cabc1ae42c46f3

* size: các size của sản phẩm

vd: size=XL-L-S

size=L

* color: các màu của sản phẩm (Những sản phẩm có màu này sẽ hiển thị)

vd: color=Do-Cam-Vang

color=XanhLa

* price: Khoảng giá

vd: price=120000-550000

* keyword: Từ khóa tìm kiếm (dựa trên tên sản phẩm)

vd: keyword=aothun

keyword=quan%20jean%20den

* sorting: cách sắp xếp danh sách sản phẩm
  + pricelowtohigh: theo giá, từ thấp đến cao
  + pricehightolow: theo giá, từ cao đên thấp
  + viewlowtohigh: theo lượt view, từ thấp đến cao
  + viewhightolow: theo lượt view, từ cao đến thấp

vd: sorting=pricelowtohigh

sorting=viewlowtohigh

### checkout.html (trang nhập thông tin thanh toán, tạo order)

#### Nội dung hiển thị

Danh sách sản phẩm trong giỏ hàng

Ô nhập thông tin nhận hàng

Nút Thanh toán

#### Tham số trong Request GET

* cartid: \_id của giỏ hàng

vd: cartid= f695ea5a-8968-41de-b6ef-65989dded391

### login.html (Trang đăng nhập)

#### Nội dung hiển thị

Tùy *type* của USER mà redirect đến trang thích hợp (index.html hoặc admin.html)

Ô đăng nhập, nút đăng nhập, nút đăng ký, nút reset mật khẩu qua email

#### Tham số trong Request GET

#### Tham số trong Request POST

* username:
* password

### register.html (Trang đăng ký)

#### Nội dung hiển thị

Đăng ký mới tài khoản.

Kiểm tra tính hợp lệ của dữ liệu trong textbox trước khi gọi hàm đăng đăng ký trên server

#### Tham số trong Request GET

#### Tham số trong Request POST

* username
* password
* name
* email
* phone
* address

### account-settings.html (Trang quản lý tài khoản)

#### Nội dung hiển thị

Xem/Cập nhật thông tin tài khoản

Kiểm tra tính hợp lệ của dữ liệu trong textbox trước khi gọi hàm cập nhật dữ liệu trên server

Lịch sử đơn đặt hàng và trạng thái của từng đơn

#### Tham số trong Request GET

* username:

vd: username=user1

username=user2

#### Tham số trong Request POST

### “./views/layouts/layout.hbs” (Layout chính trên trang chủ - Giỏ hàng)

#### Nội dung hiển thị

Các thao tác liên quan đến giỏ hàng (dùng AJAX)

Nút thanh toán (Chuyển đến trang checkout.html)

#### Tham số trong Request GET

#### Tham số trong Request POST

<Chưa biết>

### admin.html (Trang admin)

#### Nội dung hiển thị

Gồm các trang con:

* admin.html/product: Quản lý sản phẩm (Thêm, xóa, sửa, uphình),
  + Tham số trong Request POST: Chưa biết
* admin.html/user: Quản lý user (Đổi thông tin, xóa)
  + Tham số trong Request POST: Chưa biết
* admin.html/order: Quản lý đơn hàng (Xem, xóa, đổi *status*)
  + Tham số trong Request POST: Chưa biết

# Controller

“.\controller”

## business.js

Hàm trong file này được sử dụng chủ yếu ở “.\route”

Các hàm trả về dữ liệu thông qua hàm **callback**

### GetAllProduct(callback)

Gọi hàm QueryProducts() ở DAL.js

callback(products)

products: mảng **Product**

### GetProductByPageIndex(pageIndex, callback)

Gọi hàm QueryProducts() ở DAL.js

Chia danh sách **PRODUCT** trong CSDL theo trang và trả về mảng **Product** ở trang *pageIndex*

callback(products, nProducts)

products: mảng **Product** đã cắt theo trang

nProducts: Tổng số lượng **PRODUCT** có trong CSDL

### GetProductByCategoryID(categoryID, pageIndex, callback)

Gọi hàm QueryProducts() ở DAL.js

Lọc tất cả **Product** theo **Category** và trả về mảng **Product** ở trang *pageIndex*

callback(products, nProducts)

products: mảng **Product** đã cắt theo trang

nProducts: Tổng số lượng **PRODUCT** có trong CSDL

### GetAllCategory(callback)

Gọi hàm QueryProducts() ở DAL.js

Trả về mảng **Category**

callback(category)

category: Mảng **Category**

### GetRelatedProduct(productID, topN, callback)

Lấy danh sách sản phẩm liên quan

#### Tham số

* productID: String: ID của sản phẩm chính
* topN: Number: Số lượng sản phẩm liên quan nhất cần lấy
* callback: Hảm trả về kết quả
  + callback(thamso1)
    - thamso1: Mảng các **Product** tìm được

#### Thuật toán

### GetProductComments(productID, callback)

Trả về danh sách Comment của *productID*

#### Tham số

* productID: String: ID của sản phẩm chính
* callback: Hảm trả về kết quả
  + callback(thamso1)
    - thamso1: Mảng các **Comment**. Nếu không có trả về ***null***

#### Thuật toán

### GetProductCommentsByPageIndex(productID, pageIndex, callback)

Trả về danh sách Comment của *productID* theo trang thứ *pageIndex*

#### Tham số

* productID: String: ID của sản phẩm chính
* pageIndex:
* callback: Hảm trả về kết quả
  + callback(thamso1, thamso2)
    - thamso1: Mảng các **Comment**. Nếu không có trả về ***null***
    - thamso2: Number: Tổng số **COMMENT** của **PRODUCT**

#### Thuật toán

### AddProductComments(productID, username, content, callback)

Thêm comment vào productID

#### Tham số

* productID: String: ID của sản phẩm chính
* username: username của user hoặc tên do visitor nhập vào
* content: Nội dung bình luận
* callback: Hảm trả về kết quả
  + callback(thamso1)
    - thamso1: **bool**: Thêm vào thành công hay không

#### Thuật toán

### GenerateCart(sessionID, callback)

Tạo mới một **CART** có \_*id* là \_*sessionID*

#### Tham số

* sessionID: id của session hiện tại
* callback: Hảm trả về kết quả
  + callback(thamso1, thamso2)
    - thamso1: bool: Tạo thành công hay không
    - thamso2: String: \_id của CART vừa tạo

#### Thuật toán

### GetCart(cartID, callback)

Lấy giỏ hàng có \_id là *cartID*

#### Tham số

* cartID: \_id của **CART**
* callback: Hảm trả về kết quả
  + callback(thamso1, thamso2)
    - thamso1: Object **Cart**. Nếu không có trả về ***null***
    - thamso2: Number: Tổng số **CARTDETAIL**

#### Thuật toán

### AddItemToCart(cartID, productID, quantity, color, callback)

Thêm mới 1 **CARTDETAIL** vào giỏ hàng có \_id là *cartID*

#### Tham số

* cartID: \_id của **CART**
* productID: \_id của Product cần thêm vào giỏ
* quantity: số lượng sản phẩm
* color: màu của sản phẩm (“Do”, “Cam”,…)
* callback: Hảm trả về kết quả
  + callback(thamso1)
    - thamso1: bool: Thêm vào thành công hay không

#### Thuật toán

### ChangeItemInCart(cartID, itemIndex, productID, quantity, color, callback)

Update phần tử *itemIndex* của mảng **CARTDETAIL** trong **CART** *cartID*

#### Tham số

* cartID: \_id của CART
* itemIndex: chỉ số của phần tử trong mảng **CARTDETAIL** cần được update
* productID, quantity, color: Thuộc tính của **CARTDETAIL** mới
* callback: Hảm trả về kết quả
  + callback(thamso1)
    - thamso1: bool: cập nhật thành công hay không

#### Thuật toán

### RemoveItemInCart(cartID, itemIndex, callback)

Xóa phần Object **CARTDETAIL** thứ *itemIndex* trong giỏ hàng *cartID*

#### Tham số

* cartID: \_id của CART
* itemIndex: chỉ số của phần tử trong mảng **CARTDETAIL** cần được xóa
* callback: Hảm trả về kết quả
  + callback(thamso1)
    - thamso1: bool: Xóa thành công hay không

#### Thuật toán

### RegisterNewUser(type, username, password,name,email,phone,address, callback)

Tạo object **USER** và gọi hàm insert vào CSDL, sau đó gửi email chứa mã kích hoạt cho user

#### Tham số

* Type,username…: Thuộc tính của **USER**
* callback: Hàm trả về kết quả
  + callback(thamso1)
    - thamso1: bool: Thêm thành công hay không

#### Thuật toán

### GetUser(username, callback)

Lấy **User** dựa vào username

#### Tham số

* username:
* callback: Hảm trả về kết quả
  + callback(thamso1)
    - thamso1: **User** tìm được

#### Thuật toán

### ValidateLogin(username, password, callback)

Kiểm tra đăng nhập của user

#### Tham số

* Username,password: Thuộc tính của **USER**
* callback: Hảm trả về kết quả
  + callback(thamso1, thamso2)
    - thamso1: bool: Thêm thành công hay không
    - thamso2: String: Type của **User** này (‘admin’, ‘customer’)

#### Thuật toán

### ChangeUserInfo(username,name,phone,address , callback)

Đổi thông tin **User** *username*

#### Tham số

* username: Username của user cần đổi thông tin
* password, name, phone, address: Thông tin mới
* callback: Hảm trả về kết quả
  + callback(thamso1)
    - thamso1: bool: cập nhật thành công hay không

#### Thuật toán

### ChangePassword(username,newPassword, callback)

Đổi mật khẩu của **User** *username*

#### Tham số

* username: Username của user cần đổi thông tin
* newPassword: Mật khẩu mới
* callback: Hảm trả về kết quả
  + callback(thamso1)
    - thamso1: bool: cập nhật thành công hay không

#### Thuật toán

### ForgetPassword (username, callback)

Tạo mật khẩu mới và gửi email mật khẩu đó cho user

#### Tham số

* username: Username của user cần đổi thông tin
* callback: Hảm trả về kết quả
  + callback(thamso1)
    - thamso1: bool: cập nhật thành công hay không

#### Thuật toán

### Order(username, cartID, note, recipientName, address, phone, callback)

Tạo object **Order** rồi gọi hàm Create tương ứng trong CSDL

#### Tham số

* username: User đặt hàng
* cartID: \_id của **CART** tương ứng với **User** (Xác định bằng session id)
* note: Ghi chú của đơn đặt hàng
* recipientName, address, phone: Thông tin người nhận hàng
* callback: Hảm trả về kết quả
  + callback(thamso1)
    - thamso1: bool: Thêm thành công hay không

#### Thuật toán

### GetAllOrderOfUser(username, callback)

Truy vấn tất cả **ORDER** của **USER** *username*

#### Tham số

* username:
* callback: Hảm trả về kết quả
  + callback(thamso1)
    - thamso1: mảng **Order**

#### Thuật toán

### GetSaleStatistic (startDate, endDate, criteria, callback)

Thống kê doanh số theo Ngày/tuần/tháng/năm/quý (*criteria)*

#### Tham số

* startDate: Ngày bắt đầu
* endDate: Ngày kết thúc
* criteria: Tiêu chí thống kê: “ngay”,”tuan”,”thang”,”quy”,”nam”
* callback: Hảm trả về kết quả
  + callback(thamso1)
    - thamso1: mảng 1 chiều tổng doanh thu của mỗi *criteria*

#### Thuật toán

### GetTopProducts (topN, callback)

Trả về top *topN* sản phẩm có số lượng được bán nhiều nhất

#### Tham số

* topN: Số lượng phần tử top
* callback: Hảm trả về kết quả
  + callback(thamso1)
    - thamso1: mảng **Product** *topN* phần tử

#### Thuật toán

# Model

“.\model”

## DAL.js

### QueryProducts(query, callback)

Truy vấn sản phẩm trong CSDL

#### Tham số

* query: Object truy vấn gồm các thuộc tính sau:
  + category: \_id của category (cú pháp: “5b23d24993cabc1ae42c46f1”)
  + size: danh sách các size (cú pháp: “XL-S-M”, “XL”)
  + color: danh sách màu (cú pháp: “Do-Cam-XanhLa”, “Cam”)
  + price: khoảng giá (cú pháp: “12000-90000”)
  + keyword: Từ khóa tìm kiếm (So sánh với thuộc tính *name* của Product)
* callback: Hảm trả về kết quả
  + callback(thamso1)
    - thamso1: Mảng các **Product** tìm được

#### Thuật toán

Bước 1: Xử lý object query

Chuyển các thuộc tính sang đúng với cú pháp truy vấn dùng trong hàm find(), update(), insert()… của mongoose

Bước 2: Truy vấn CSDL

Gọi hàm find(), update(), insert() với object query đã xử lý

Bước 3: Xử lý mảng vừa truy vấn được

Bước 4: Return mảng đã xử lý

### QueryRelatedProducts(productID, topN, callback)

Truy vấn top *topN* danh sách các sản phẩm liên quan của sản phẩm *productID*

#### Tham số

* productID: String: ID của sản phẩm chính
* topN: Number: Số lượng sản phẩm liên quan nhất cần lấy
* callback: Hảm trả về kết quả
  + callback(thamso1)
    - thamso1: Mảng các **Product** tìm được

#### Thuật toán

### QueryProductComments(productID, callback)

Trả về danh sách **Comment** của *productID*

#### Tham số

* productID: String: ID của sản phẩm chính
* callback: Hảm trả về kết quả
  + callback(thamso1, thamso2)
    - thamso1: Mảng các **Comment**. Nếu không có trả về *null*
    - thamso2: Number: Tổng số **COMMENT**

#### Thuật toán

### InsertProductComments(productID, comment, callback)

Thêm comment vào productID

#### Tham số

* productID: String: ID của sản phẩm chính
* comment: Object COMMENT (Như mô tả CSDL)
* callback: Hảm trả về kết quả
  + callback(thamso1)
    - thamso1: bool: Thêm vào thành công hay không

#### Thuật toán

### CreateCart(sessionID, callback)

Tạo mới một **CART** có \_*id* là \_*sessionID*

#### Tham số

* sessionID: id của session hiện tại
* callback: Hảm trả về kết quả
  + callback(thamso1, thamso2)
    - thamso1: bool: Tạo thành công hay không
    - thamso2: String: \_id của CART vừa tạo

#### Thuật toán

### QueryCart(cartID, callback)

Lấy giỏ hàng (danh sách CARTDETAIL) *cartID*

#### Tham số

* cartID: \_id của CART
* callback: Hảm trả về kết quả
  + callback(thamso1, thamso2)
    - thamso1: Object CART. Nếu không có trả về *null*
    - thamso2: Number: Tổng số CARTDETAIL

#### Thuật toán

### InsertItemToCart(cartID, newCartDetail, callback)

Thêm 1 CARTDETAIL mới vào CART *cartID*

#### Tham số

* cartID: \_id của CART
* newCartDetail: Object CARTDETAIL cần thêm vào giỏ hàng
* callback: Hảm trả về kết quả
  + callback(thamso1)
    - thamso1: bool: Thêm vào thành công hay không

#### Thuật toán

### UpdateItemInCart(cartID, itemIndex, newCartDetail, callback)

Update phần tử *itemIndex* của mảng detail trong CART *cartID* bằng Object *newCartDetail*

#### Tham số

* cartID: \_id của CART
* itemIndex: chỉ số của phần tử trong mảng CARTDETAIL cần được update
* newCartDetail: Object CARTDETAIL update
* callback: Hảm trả về kết quả
  + callback(thamso1)
    - thamso1: bool: cập nhật thành công hay không

#### Thuật toán

### DeleteItemInCart(cartID, itemIndex, callback)

Xóa CARTDETAIL thứ *itemIndex* trong giỏ hàng cartID

#### Tham số

* cartID: \_id của CART
* itemIndex: chỉ số của phần tử trong mảng CARTDETAIL cần được xóa
* callback: Hảm trả về kết quả
  + callback(thamso1)
    - thamso1: bool: Xóa thành công hay không

#### Thuật toán

### CreateUser(newUser, callback)

Thêm object USER *newUser* vào CSDL

#### Tham số

* newUser: Object User mới cần thêm (đã đúng chuẩn trong CSDL)
* callback: Hảm trả về kết quả
  + callback(thamso1)
    - thamso1: bool: Thêm thành công hay không

#### Thuật toán

### QueryUser(username, callback)

Truy vấn user dựa vào username.

Nếu username = “” trả về tất cả User

#### Tham số

* username:
* callback: Hảm trả về kết quả
  + callback(thamso1)
    - thamso1: User tìm được

### UpdateUser(username, newUser, callback)

Cập nhật thông tin theo username dựa trên newUser

#### Tham số

* username:
* newUser: Object User cấu trúc như mô tả trong CSDL
* callback: Hảm trả về kết quả
  + callback(thamso1)
    - thamso1: Cập nhật thành công

#### Thuật toán

### CreateOrder(newOrder, callback)

Tạo ORDER trong CSDL từ newOrder

#### Tham số

* newOrder: Object ORDER
* callback: Hảm trả về kết quả
  + callback(thamso1)
    - thamso1: bool: Thêm thành công hay không

#### Thuật toán

### QueryOrder(username, callback)

Truy vấn tất cả ORDER của USER *username*

Nếu username = “” trả về toàn bộ ORDER

#### Tham số

* username:
* callback: Hảm trả về kết quả
  + callback(thamso1)
    - thamso1: mảng ORDER

#### Thuật toán