**TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI**

VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG

---------------o0o---------------

**BÁO CÁO MÔN HỌC   
PROJECT III**

----------

***Xây dựng Website thời trang***

Giảng viên hướng dẫn: TS. Nguyễn Kim Khánh

Sinh viên thực hiện: Đỗ Văn Cường – 20172986

**Hà Nội, Tháng 01 năm 2021**

**MỤC LỤC**

[LỜI CẢM ƠN 3](#_Toc59028351)

[TÀI LIỆU THAM KHẢO 3](#_Toc59028352)

[LỜI NÓI ĐẦU 4](#_Toc59028353)

[**I.** **KHẢO SÁT, ĐẶC TẢ YÊU CẦU BÀI TOÁN** 5](#_Toc59028354)

[**1.** **Mô tả yêu cầu bài toán** 5](#_Toc59028355)

[**2.** **Mô tả tổng quan** 5](#_Toc59028356)

[**II.** **PHÂN TÍCH THIẾT KẾ HỆ THỐNG** 6](#_Toc59028357)

[1. **Biểu đồ Use case tổng quan** 6](#_Toc59028358)

[**2.** **Thiết kế cơ sở dữ liệu** 7](#_Toc59028359)

[**III.** **CÔNG NGHỆ SỬ DỤNG** 7](#_Toc59028360)

[**1.** **Front-end** 7](#_Toc59028361)

[**2.** **Back-end** 8](#_Toc59028362)

[**3.** **Database** 8](#_Toc59028363)

[**IV.** **PHÂN CHIA CÔNG VIỆC** 9](#_Toc59028364)

[**V.** **KẾT QUẢ THỰC HIỆN** 9](#_Toc59028365)

[**1.** **Một số hình ảnh minh họa sản phẩm** 9](#_Toc59028366)

[**2.** **Kết quả** 11](#_Toc59028367)

[**VI.** **KẾT LUẬN** 12](#_Toc59028368)

# 

# LỜI CẢM ƠN

Chúng em xin được bày tỏ sự kính trọng và biết ơn đối với giảng viên TS. Nguyễn Kim Khánh, Viện Công nghệ thông tin và truyền thông, trường Đại học Bách khoa Hà Nội. Thầy đã tận tình chỉ bảo, hướng dẫn giúp chúng em thêm những kiến thức trong môn học Project III. Những nền tảng cơ sở thầy đã truyền dạy cho chúng em đã giúp cho chúng em hoàn thành bài tập lớn này. Chúng em xin chân thành cảm ơn thầy.

Hà Nội, tháng 12 năm 2020

Nhóm sinh viên thực hiện:

Đỗ Văn Cường

# TÀI LIỆU THAM KHẢO

1. TS. Nguyễn Kim Khánh, Đại Học Bách Khoa Hà Nội, 2020
2. Tham khảo các hệ thống tương tự:

* <https://gomart.com.vn/>
* <https://www.beemart.vn/>
* <https://vinshop.vn/>
* <https://shopee.vn/>

# LỜI NÓI ĐẦU

Trong thời đại mới, xu thế mua hàng online ngày càng lan rộng trong tất cả các tầng lớp trong xã hội. Đại dịch Covid-19 cho thấy một bước chuyển mình mạnh mẽ của các doanh nghiệp bán lẻ, đó là cuộc đua bán hàng trực tuyến bằng công nghệ 4.0 với dịch vụ “mua bán online - giao hàng tận nhà”. Nhiều chuỗi cửa hàng như Shopee, Tiki, Lazada… cũng đã tung ra các sàn thương mại điện tử ship hàng tận nơi. Tuy nhiên còn một bộ phận lớn những cửa hàng nhỏ không có tiềm lực nhân viên để có thể giao tận nhà cho từng khách hàng như các chuỗi siêu thị, các hệ thống bán lẻ lớn.

Trước tình hình đó, việc có một hệ thống hỗ trợ cho các cửa hàng tạp hóa nhỏ lẻ có thể triển khai dịch vụ quản lý sản phẩm, đặt hàng online là rất cần thiết.

Chính vì vậy, em đã quyết định chọn đề tài *Xây dựng Website quản lý cửa hàng thời trang và bán, giao hàng online.* Website chúng em xây dựng nhằm đưa sản phẩm trực tiếp tới tận tay người tiêu dùng. Thông qua hệ thống, người tiêu dùng có thể dễ dàng chọn mua được đồ mong muốn ở cửa hàng và có người mang về tận nhà mà không cần phải trực tiếp đi đến cửa hàng.

Trong quá trình phát triển Website, chúng em chắc chắn còn có nhiều thiếu sót trong các nghiệp vụ chức năng cũng như các lỗi khác, em rất mong nhận được sự góp ý từ thầy và các bạn để ngày càng hoàn thiện hơn. Em xin chân thành cảm ơn!

1. **YÊU CẦU CHỨC NĂNG**
2. **Mô tả yêu cầu bài toán**

Người dùng có thể mua hàng trực tiếp tại trang web hoặc tới cửa hàng mua trực tiếp. Từ đó hệ thống sẽ xác nhận đơn đặt hàng của người tiêu dùng, tiến hành xác nhận đơn hàng cho người mua và giao hàng tại nhà cho người mua theo địa chỉ người dùng đặt hàng.

Website được xây dựng đầy đủ những chức năng cơ bản cho người tiêu dùng và quản trị viên.

Giao diện hệ thống Website thân thiện, dễ tương tác. Các công nghệ phải đảm bảo tính nhất quán, rõ ràng, tiện lợi cho việc nâng cấp, bảo trì sau này.

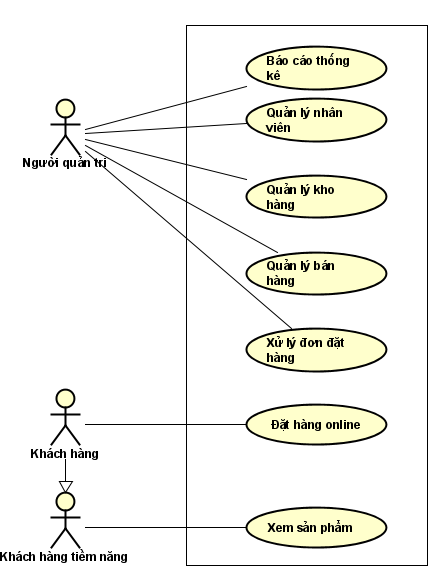
1. **Mô tả tổng quan**
   1. Các tác nhân

+ Người dùng

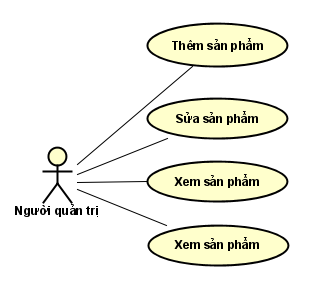
+ Người quản trị

* 1. Quy trình xử lý cơ bản
* Người tiêu dùng truy cập vào Website và đăng ký tài khoản người dùng. Hệ thống sẽ lưu lại thông tin tài khoản hợp lệ của người dùng. Sau khi đăng ký hợp lệ tài khoản, người tiêu dùng có thể đăng nhập được vào hệ thống.
* Sau khi đăng nhập thành công vào hệ thống, người dùng có thể: Thêm các thông tin tài khoản cá nhân, sửa thông tin tài khoản, tìm kiếm các cửa hàng quan tâm, tìm kiếm các mặt hàng quan tâm… trên hệ thống và thêm sản phẩm vào giỏ hàng. Hệ thống hiển thị ra những thông tin theo yêu cầu của người tiêu dùng.
* Sau khi thêm sản phẩm cần mua vào giỏ hàng, người tiêu dùng có thể tiến hành đặt hàng. Để có thể đặt hàng, người dùng cần phải điền đẩy đủ thông tin cá nhân như tên, số điện thoại, địa chỉ,... hợp lệ và xác nhận đặt hàng. Sau khi xác nhận thông tin hợp lệ thì hệ thống xác nhận đơn hàng thành công. Hệ thống xác nhận đơn và tiến hành gửi thông tin đến quản trị viên, quản trị viên sẽ xác nhận đơn hàng và tiến hành đóng gói để giao hàng

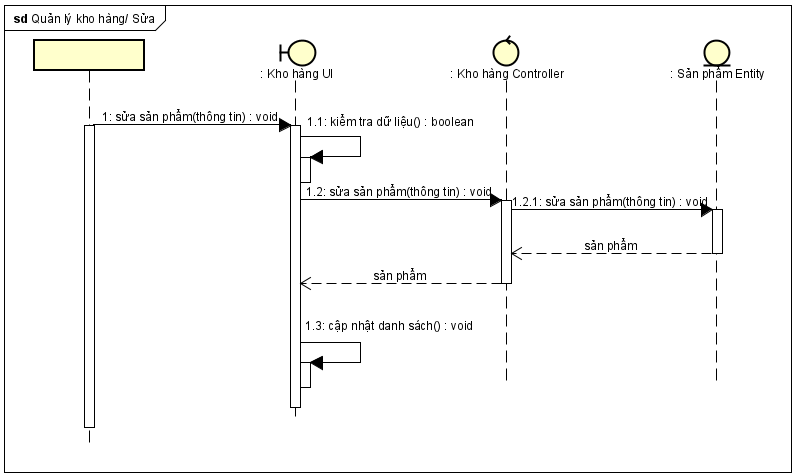
1. **PHÂN TÍCH THIẾT KẾ HỆ THỐNG**
2. **Phân tích thiết kế**
   1. **Biểu đồ use case tổng quan**

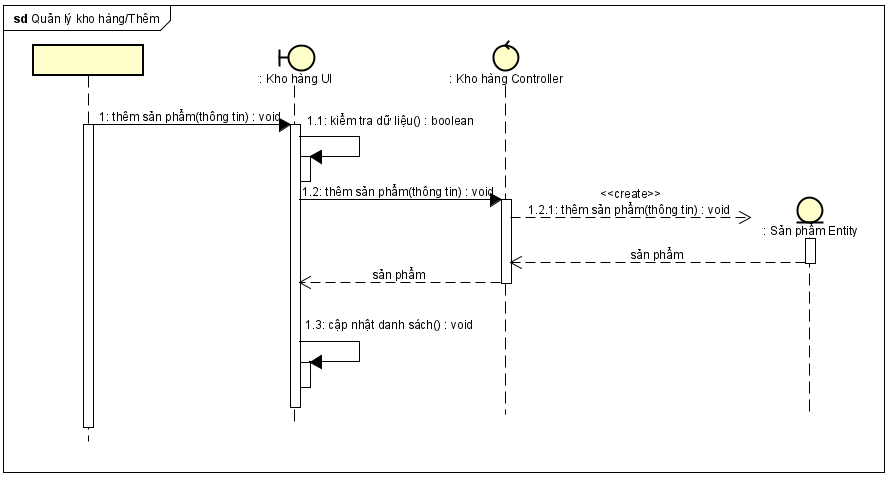


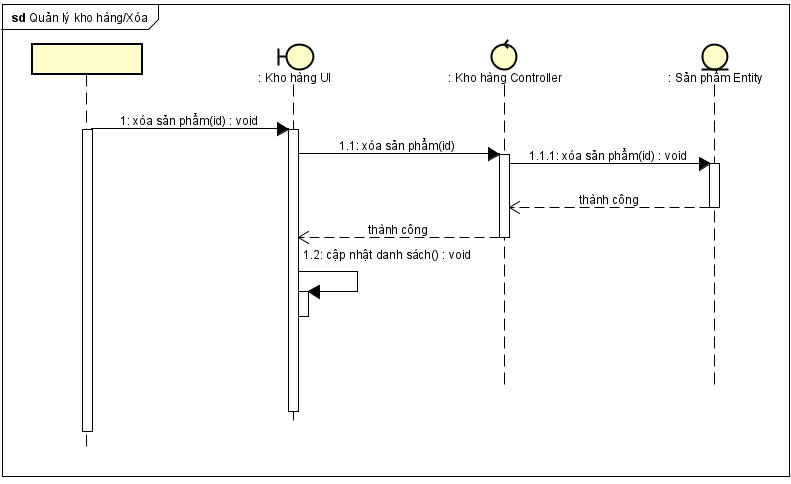
* 1. **Biểu đồ use case quản lý kho**

****

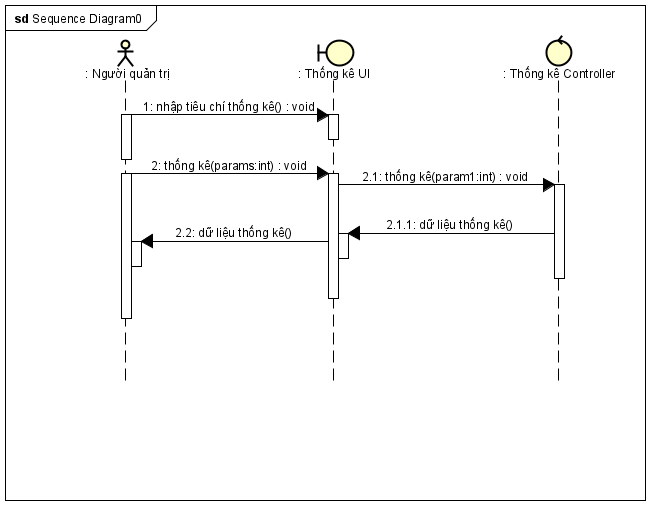
* 1. **Biểu đồ trình tự quản lý kho khàng**

****

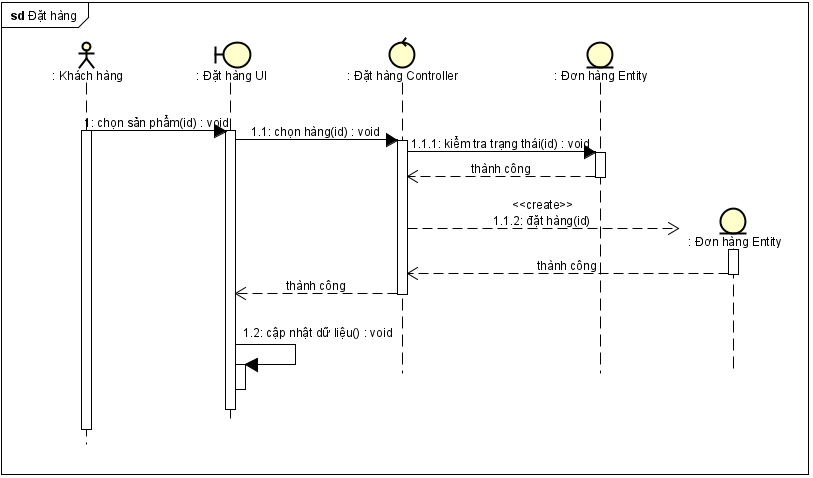
****

****

* 1. **Biểu đồ trình tự thống kê**

****

* 1. **Biểu đồ trình tự đặt hàng**

****

1. **Thiết kế cơ sở dữ liệu**
   1. **Bảng Order**

const mongoose = require('mongoose');

const orderSchema = new mongoose.Schema(

    {

        status: {

            type: String,

           enum: ['finished','completed','confirmed','unconfirmed','cart'],

            default: 'unconfirmed',

        }, //trạng thái của đơn hàng, có 5 trạng thái

        user: {

            type: mongoose.Schema.Types.ObjectId,

            ref: 'User',

        }, //id người dùng đặt hàng

        totalPrice: {

            type: Number,

        }, //tổng giá tiền của đơn hàng

        order: [

            {

                product: {

                    type: mongoose.Schema.Types.ObjectId,

                    ref: 'Product',

                }, //id sản phẩm được đặt hàng

                productID: String, //mã của sản phẩm được đặt hàng

                amount: Number, //tổng số lượng của từng sản phẩm

            },

        ], //mảng các sản phẩm được đặt hàng

    },

    {

        // Make Mongoose use Unix time (seconds since Jan 1, 1970)

        timestamps: { currentTime: () => Date.now() },

    }

);

orderSchema.set('timestamps', true);

module.exports = mongoose.model('Order', orderSchema);

* 1. **Bảng Product**

const mongoose = require('mongoose');

const productSchema = new mongoose.Schema({

    productID: {

        type: String,

        required: true,

    }, //mã sản phẩm

    name: {

        type: String,

        required: true,

        min: 0,

        max: 255,

    }, //tên sản phẩm

    description: {

        type: String,

        required: false,

        max: 1023,

        min: 0,

    }, //mô tả của sản phẩm

    status: {

        type: Boolean,

        required: true,

    }, // trạng thái bán/không bán của sản phẩm

    price: {

        type: Number,

        required: true,

    }, //giá của sản phẩm

    photo: {

        type: String,

        required: false,

    }, //ảnh của sản phẩm

    type: {

        type: mongoose.Schema.Types.ObjectId,

        ref: 'Type',

    }, //loại của sản phẩm, liên kết với model Type

    book: [

        {

            time: {

                type: Date,

                required: true,

                default: Date.now,

            }, //thời gian đặt sản phẩm

            amount: {

                type: Number,

                default: 0,

            }, //số lượng của sản phẩm được đặt trong thời gian trên

        },

    ], //mảng lưu các thông tin sản phẩm đã bán, dùng để thống kê

});

productSchema.set('timestamps', true);

module.exports = mongoose.model('Product', productSchema);

* 1. **Bảng Type**

const mongoose = require('mongoose');

const typeSchema = new mongoose.Schema({

    name: {

        type: String,

        required: true,

    }, //tên của thể loại danh mục của sản phẩm

    description: {

        type: String,

        required: false,

    }, //mô tả loại sản phẩm

    status: {

        type: Boolean,

        required: true,

        default: true,

    }, //trạng thái của loại sản phẩm dùng/không dùng

    photo: {

        type: String,

        required: false,

    }, //ảnh của loại sản phẩm

});

typeSchema.set('timestamps', true);

module.exports = mongoose.model('Type', typeSchema);

* 1. **Bảng User**

const mongoose = require('mongoose');

const userSchema = new mongoose.Schema({

    name: {

        type: String,

        required: true,

        min: 6,

        max: 30,

    }, //tên người dùng

    email: {

        type: String,

        required: true,

        max: 255,

        min: 6,

    }, //địa chỉ email của người dùng

    phone: {

        type: String,

        required: false,

    }, //số điện thoại của người dùng

    birthday: {

        type: String,

    }, //ngày sinh của người dùng

    gender: {

        type: String,

        enum: ['male', 'female', 'secret'],

        required: true,

        default: 'secret',

    }, //giới tính của người dùng, mặc định là secret

    address: {

        type: String,

        required: true,

    }, //địa chỉ của người dùng

    password: {

        type: String,

        required: true,

        max: 255,

        min: 6,

    }, //mật khẩu của người dùng

    role: {

        type: String,

        enum: ['admin', 'customer'],

        default: 'customer',

    }, //vai trò của người dùng trong hệ thống

    photo: String, //ảnh của người dùng

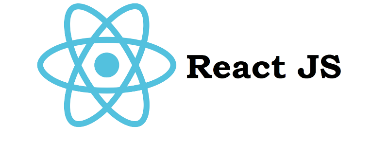
});

userSchema.set('timestamps', true);

module.exports = mongoose.model('User', userSchema);

1. **CÔNG NGHỆ SỬ DỤNG**
2. **Front-end**

Để tạo nên một trang web có giao diện đẹp, dễ sử dung, thân thiện với người dùng, Front-end nhóm em sử dụng những ngôn ngữ:

* HTML5 
* CSS3 
* React JS 

* JavaScript 

1. **Back-end**

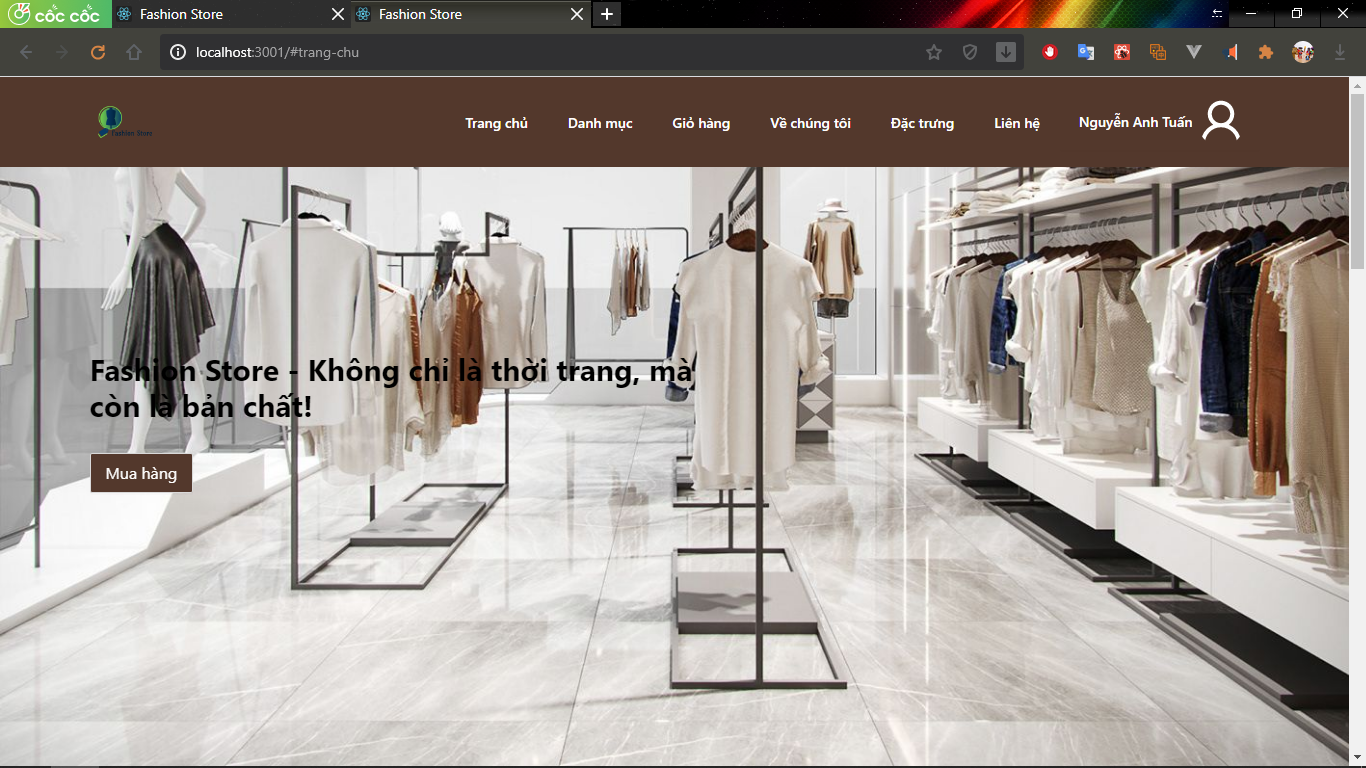
* NodeJS 

1. **Database**

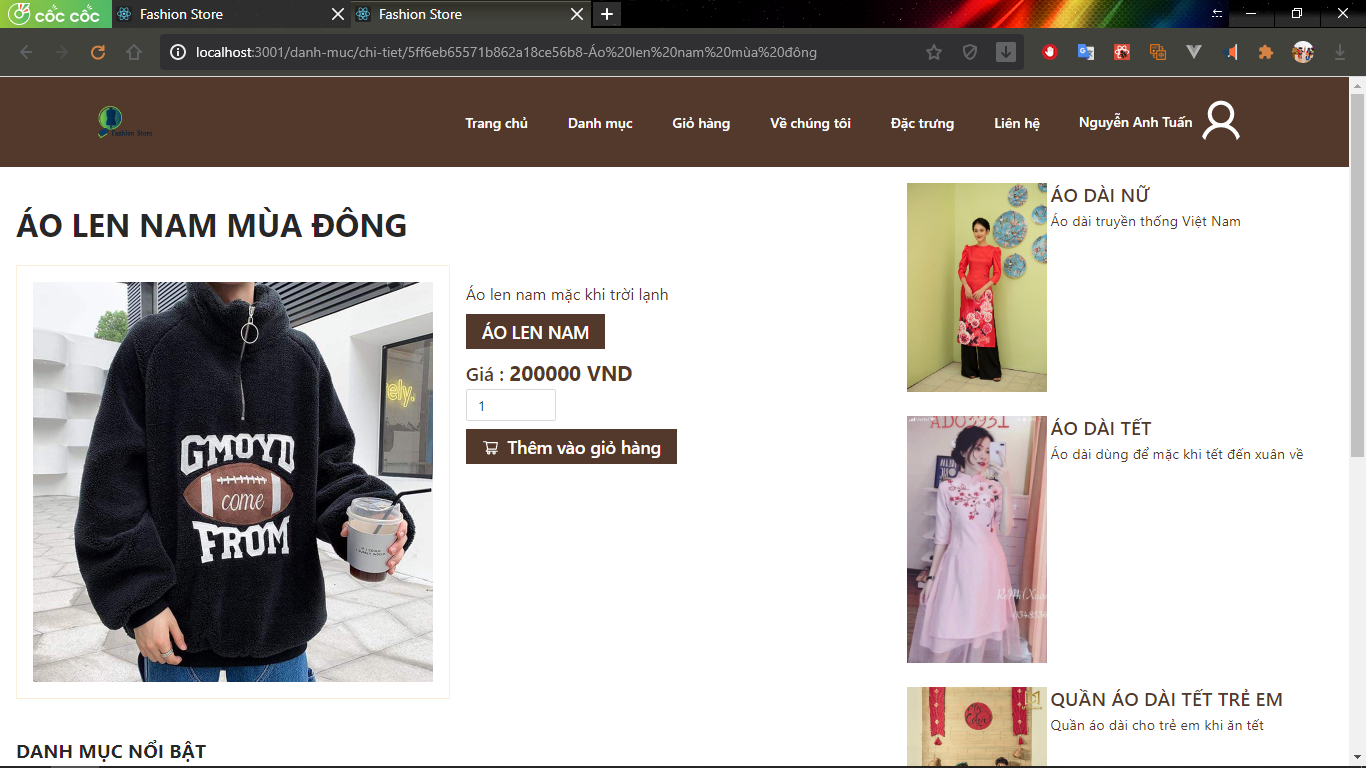
Để quản lý cơ sở dữ liệu, em lựa chọn MongoDB

* MongoDB 

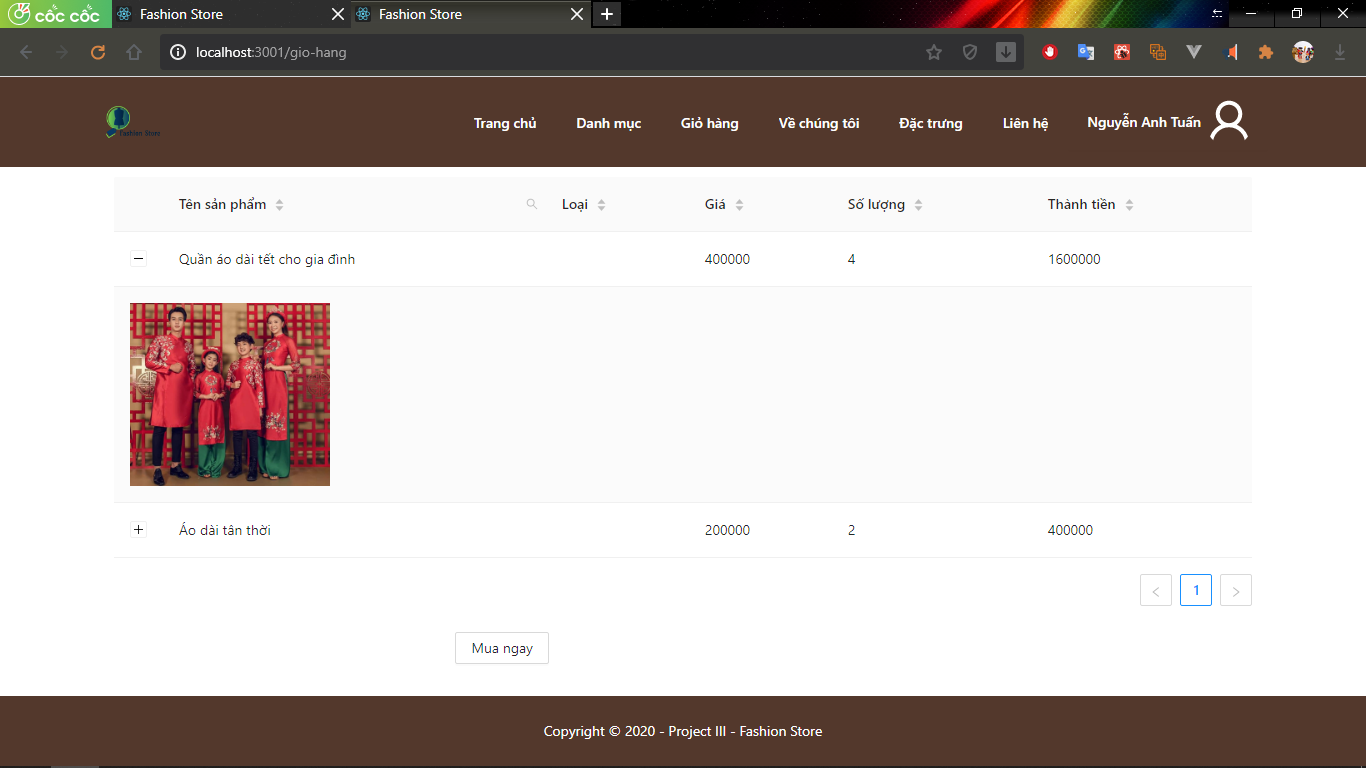
1. **KẾT QUẢ THỰC HIỆN**
2. **Một số hình ảnh minh họa sản phẩm**
   1. Giao diện trang chủ đối với tác nhân người dung



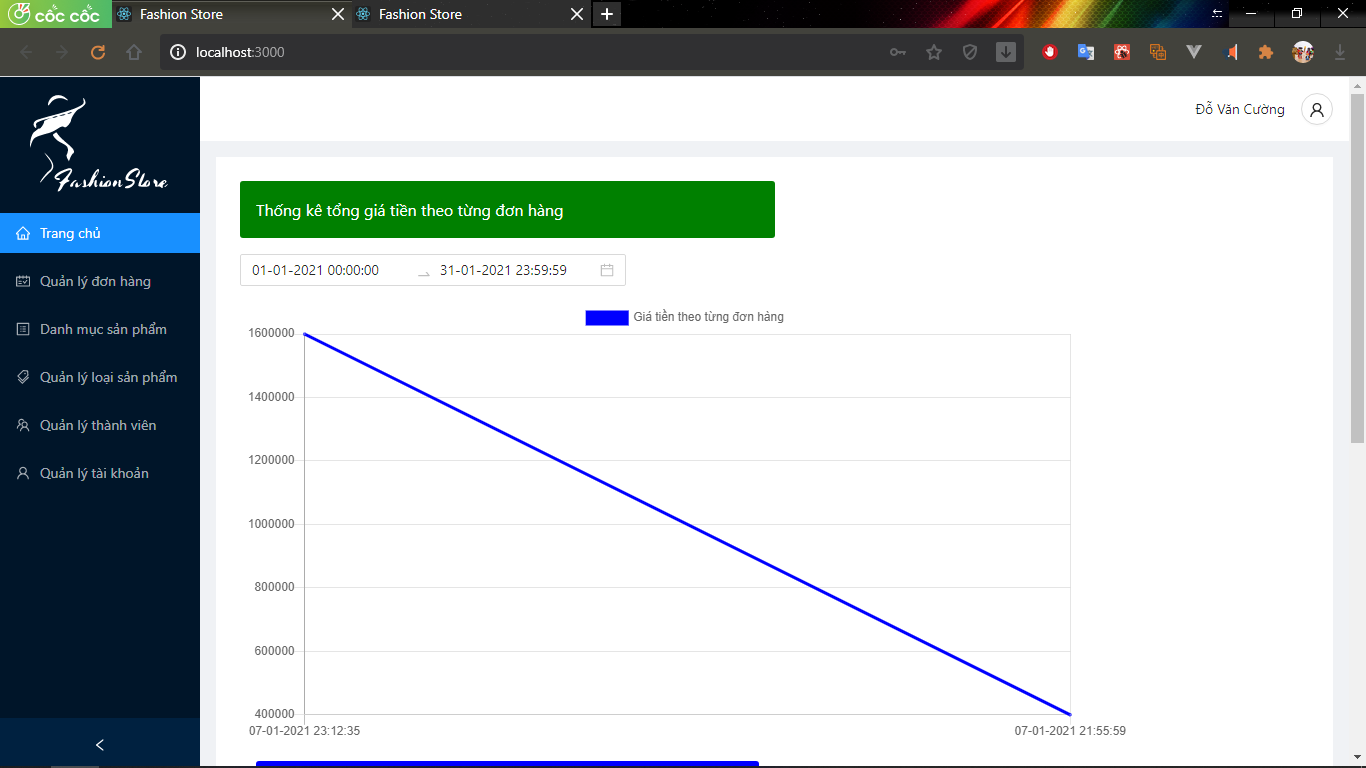
* 1. Giao diện xem chi tiết sản phẩm đối với tác nhân người dùng

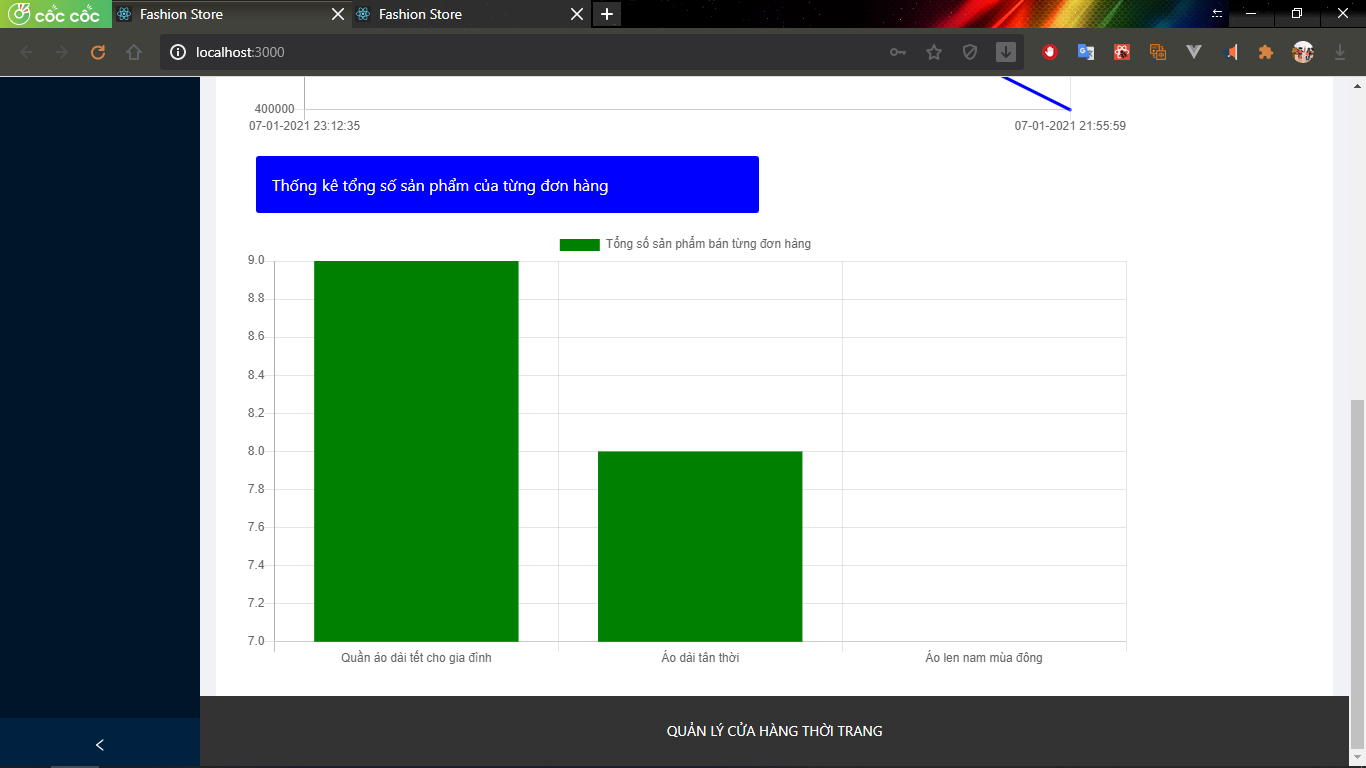


* 1. Giao diện xem giỏ hàng đối với tác nhân người dùng

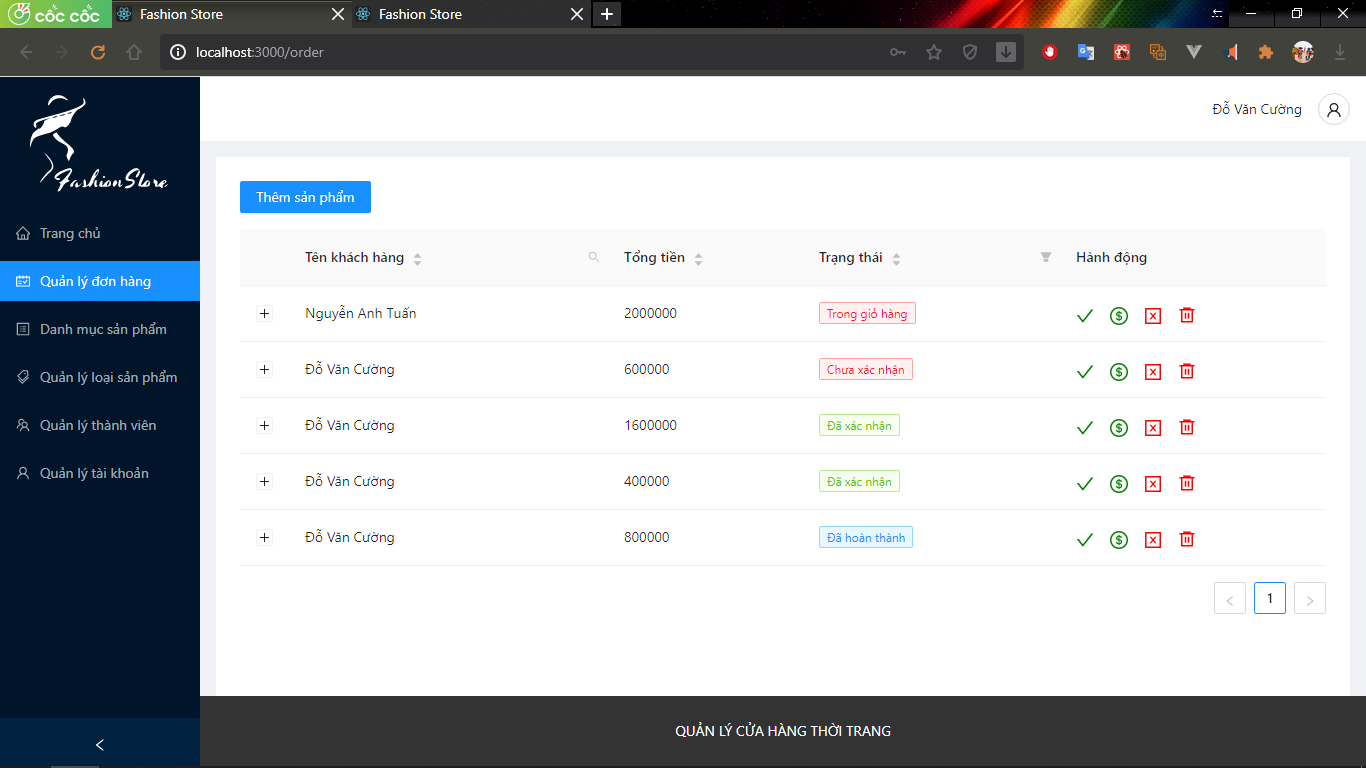


* 1. Giao diện trang chủ đối với tác nhân người quản trị

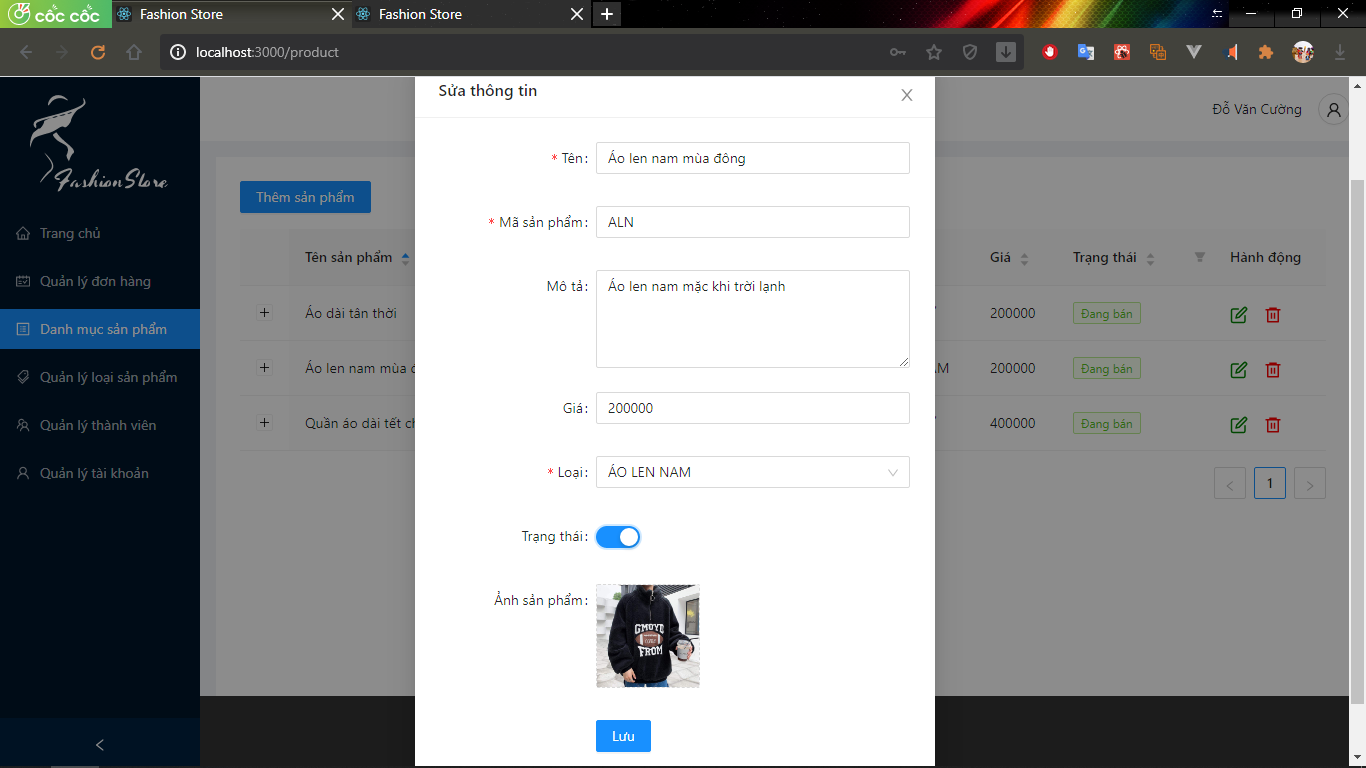




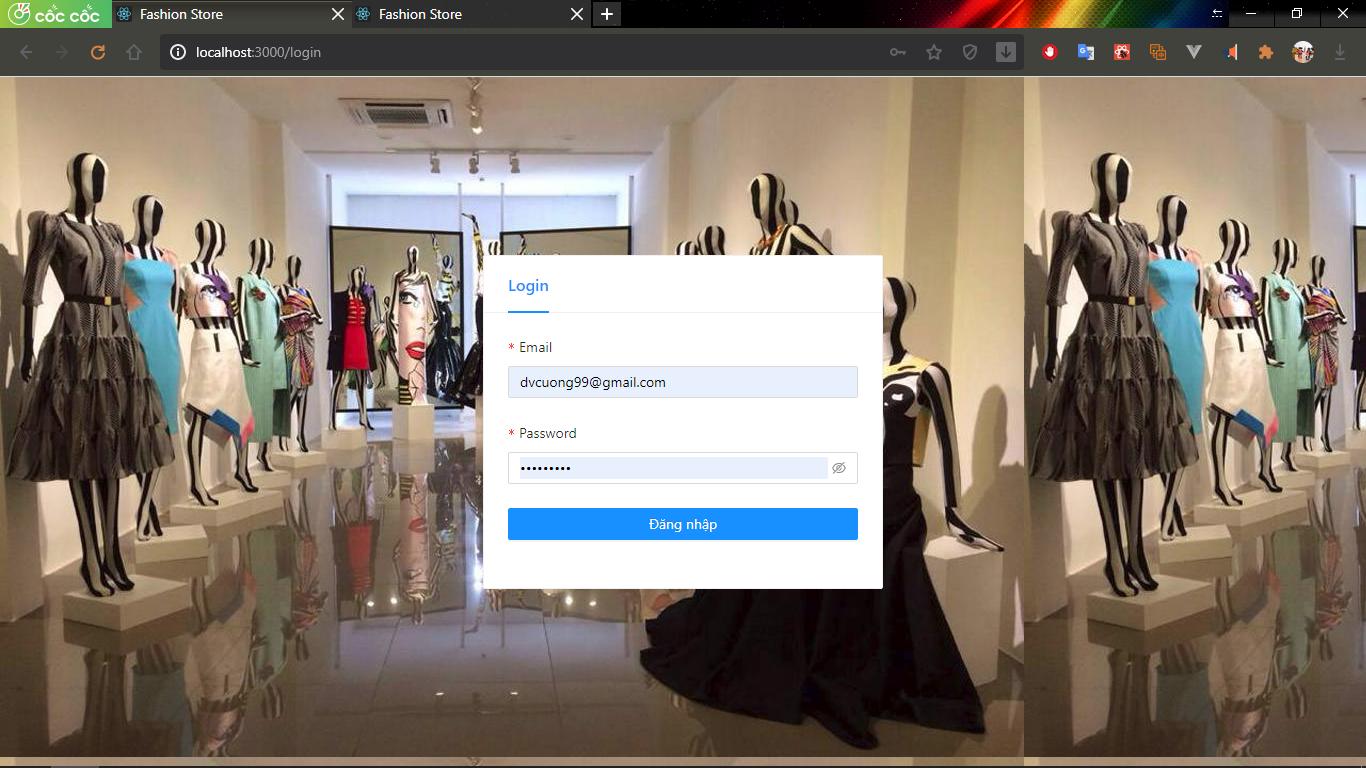
* 1. Giao diện quản lý đơn hàng



* 1. Giao diện sửa danh mục sản phẩm



* 1. Giao diện màn hình đăng nhập



1. **Kết quả**

Sau một quá trình làm việc, em đã xây dựng được một website thời trang trực tuyến, hoàn thiện cơ bản những chức năng cần thiết của một trang web để phục vụ cho quá trình mua hàng online hiện nay.

Tuy nhiên do thời gian có hạn, em chưa có nhiều kinh nghiệm thực tế nên các chức năng còn chưa thật sự hoàn thiện, chưa đảm bảo tối ưu được cho người dùng.

1. **KẾT LUẬN**

Bài tập lớn vừa là cơ hội, vừa là thách thức đối với em.

Sau khi hoàn thành bài tập lớn này, em đã có nhiều bài học kinh nghiệm quý giá. Bài tập lớn này không chỉ mang lại những bài học về kỹ năng chuyên ngành mà còn giúp em có thêm những bài học về kỹ năng mềm quý giá như việc làm việc nhóm, quản lý thời gian.