

ĐẠI HỌC BÁCH KHOA HÀ NỘI
TRƯỜNG CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG



BÁO CÁO

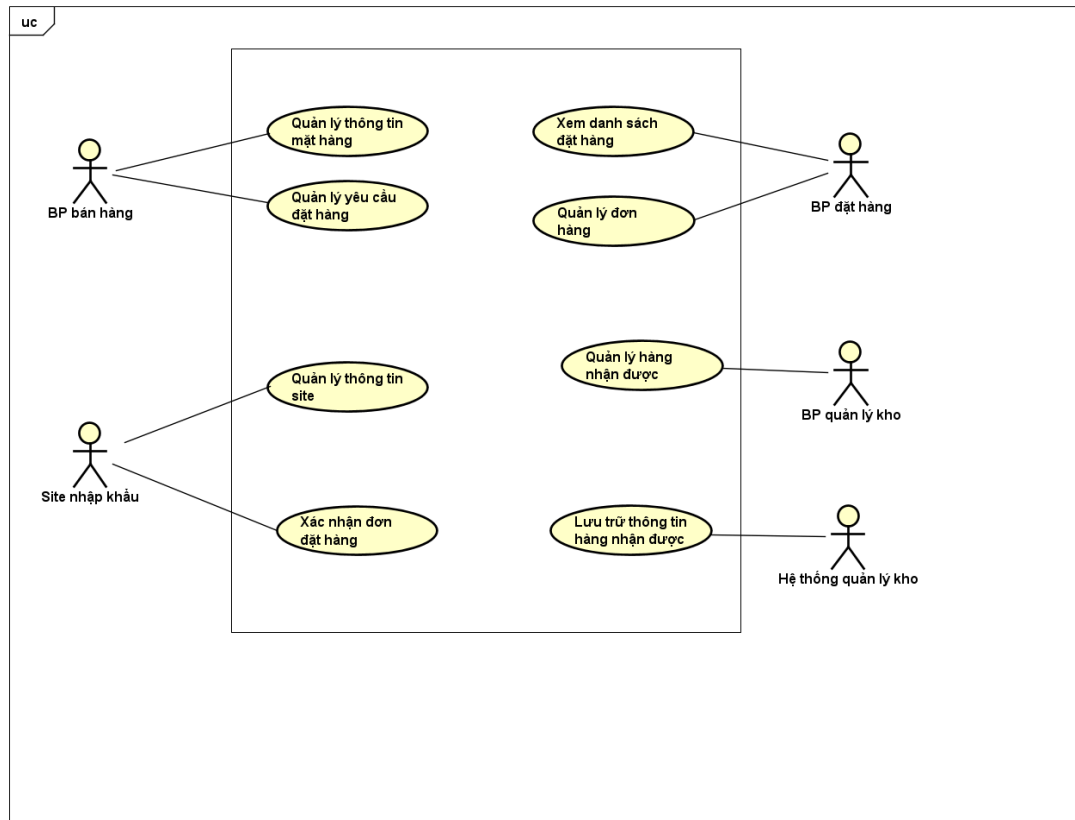
Phát triển phần mềm theo chuẩn kỹ năng ITSS

Họ và tên:	Hà Đình Nam
MSSV:	20215095
Giảng viên hướng dẫn:	TS. Trịnh Tuấn Đạt

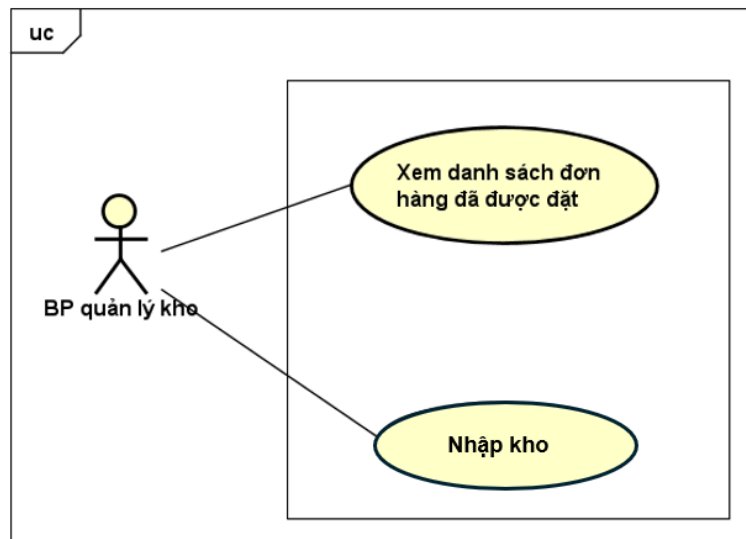
Hà Nội, 5/2024

Bài 2

Biểu đồ use case tổng quan



Phân rã



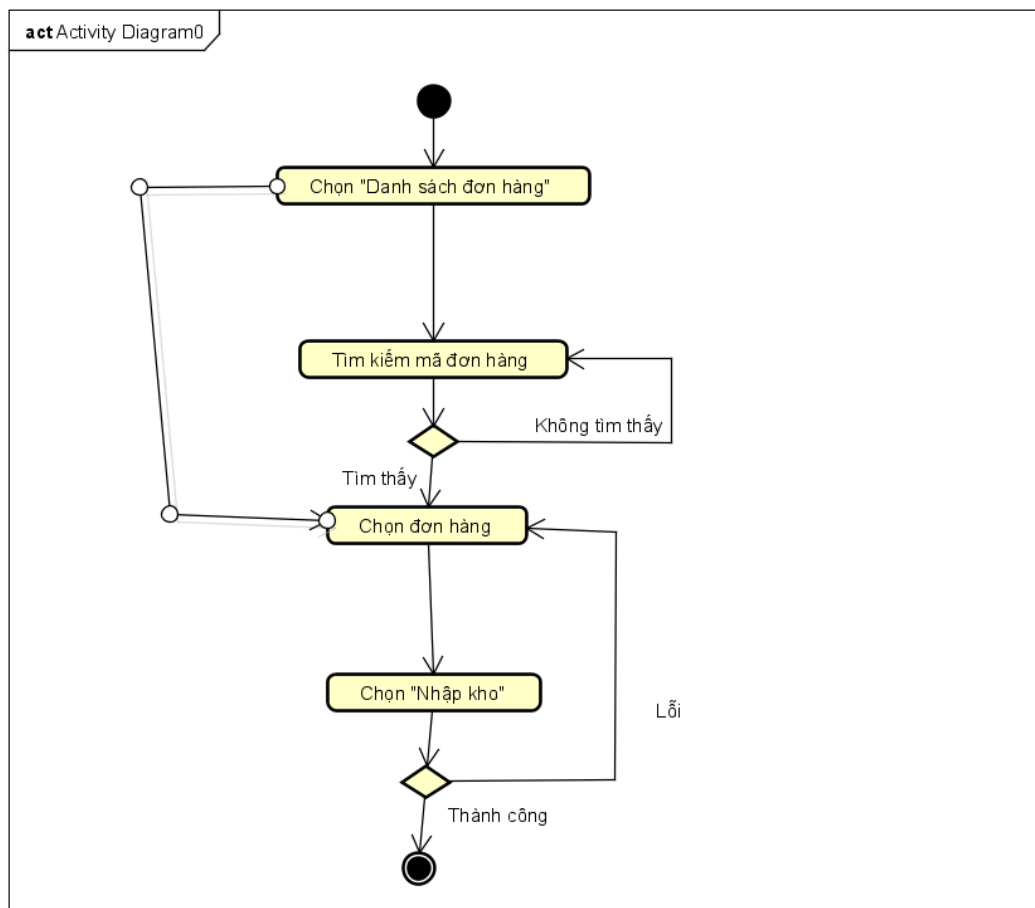
Đặc tả use case UC003 “Nhập kho”

Mã Use case	UC003	Tên Use case	Nhập kho
Tác nhân	BP quản lí kho		
Tiền điều kiện	Cần có đơn hàng giao.		
Luồng sự kiện chính (Thành công)	STT	Thực hiện bởi	Hành động
	1	BPQLK	Vào mục danh sách đơn hàng.
	2	Hệ thống	Hiển thị danh sách đơn hàng theo thời gian đặt hàng.
	3	BPQLK	Tìm kiếm đơn hàng dựa theo từ khóa, id.
	4	Hệ thống	Hiển thị các đơn hàng phù hợp.
	5	BPQLK	Chọn đơn hàng.
	6	Hệ thống	Hiển thị thông tin đơn hàng
	7	BPQLK	Nhấn nút nhập kho
	8	Hệ thống	Hiển thị thông báo nhập kho thành công
Luồng sự kiện thay thế	STT	Thực hiện bởi	Hành động
	8a.	Hệ thống	Thông báo “Đơn hàng đã được nhập trước đó!” nếu đơn hàng đó đã được nhập.

	8b	Hệ thống	Thông báo “Đơn chưa giao, không thể nhập kho!” nếu đơn hàng trạng thái đơn hàng chưa thể nhập kho.
Hậu điều kiện	Status chuyển thành “Đã nhập kho”, số lượng trong kho tăng.		

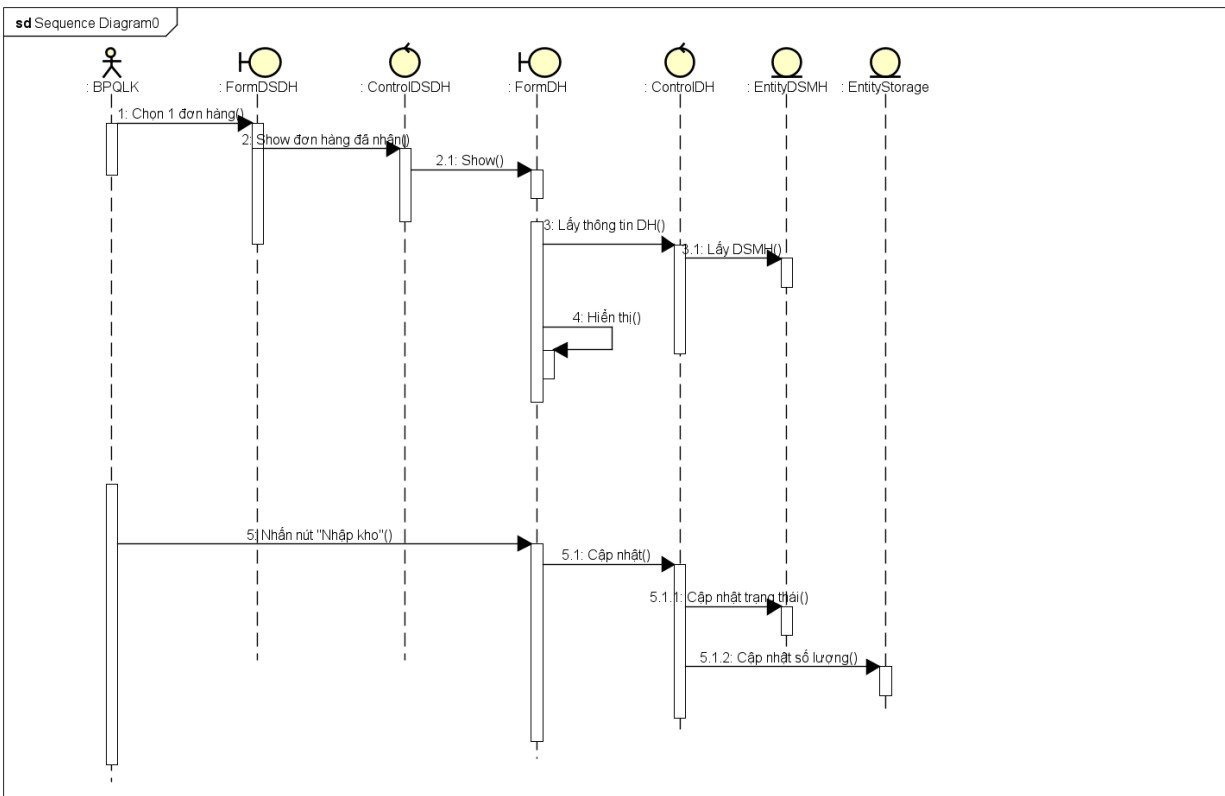
STT	Trường dữ liệu	Mô tả	Bắt buộc?	Điều kiện hợp lệ	Ví dụ
1.	Mã đơn hàng		Không		AB123Y454

2.3.2 Biểu đồ hoạt động

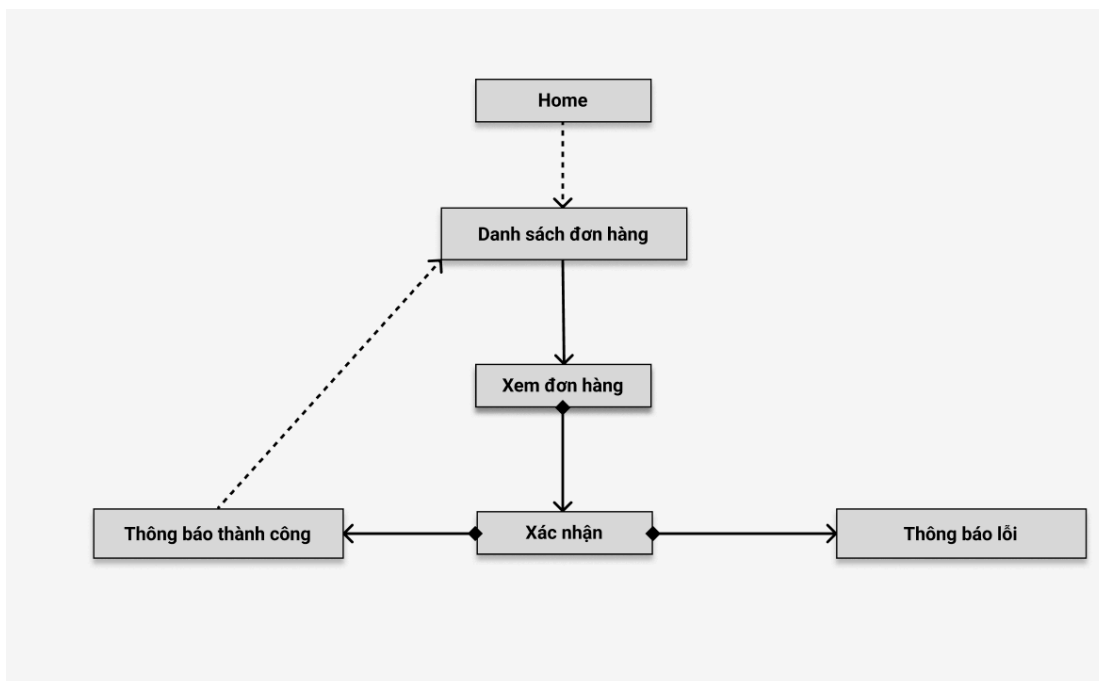


Bài 3

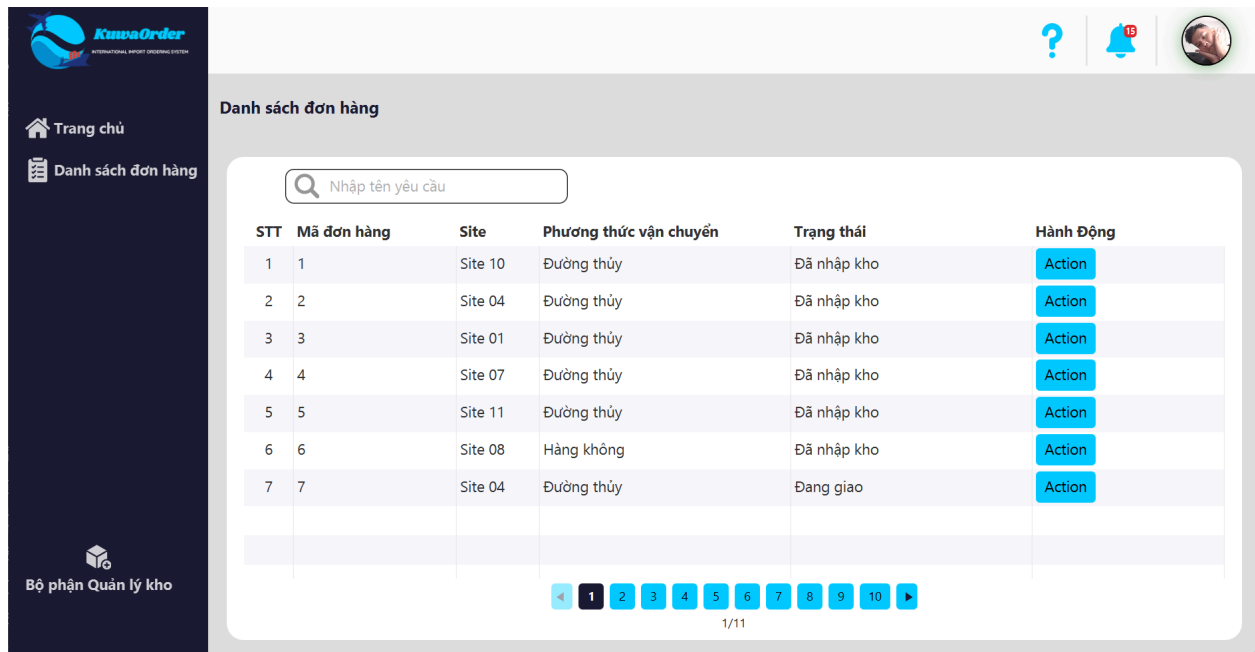
- Biểu đồ trình tự:

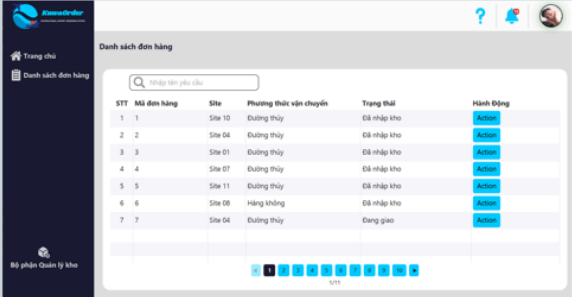


Bài 4 Sơ đồ chuyển màn



- Đặc tả màn hình
 1. Màn hình xem danh sách đơn hàng



Hệ thống đặt hàng nhập khẩu	Date of creation	Approved by	Reviewed by	Persion in charge
Screen specification	DS Đơn Hàng			Ha Dinh Nam
	Control	Operation	Function	
	Khu vực hiển thị danh sách đơn hàng chờ nhập kho	Initial	Hiển thị danh sách	
	Các nút Action	Click	Xem chi tiết đơn hàng	
	Thanh tìm kiếm	Click	Tìm kiếm	
	Thanh paginator	Click	Điều hướng giữa các trang	
	Nút Avatar	Click	Tùy chọn quản lý tài khoản	

2. Màn hình chi tiết đơn hàng

Hệ thống đặt hàng nhập khẩu		Date of creation	Approved by	Reviewed by	Persion in charge
Screen specification	Xem DH				Ha Dinh Nam
			Control	Operation	Function
			Vùng hiển thị thông tin mặt hàng	Initial	Hiển thị thông tin mặt hàng
			Nút nhập kho	Click	Nhập dữ liệu vào kho



Mã đơn hàng: 7 **Đang giao**

Tên sản phẩm: Máy hút bụi

Số lượng mua: 44 cái

Mã đơn đặt: 7

Phương thức vận chuyển: Đường thủy **Giá:** 320,750,000

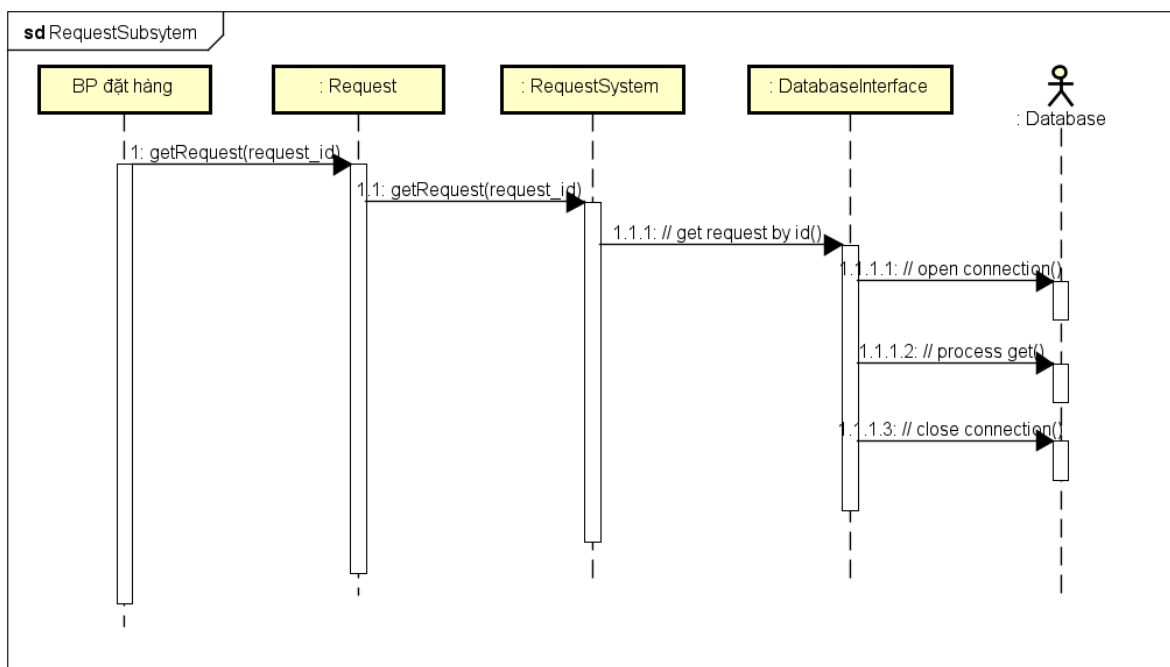
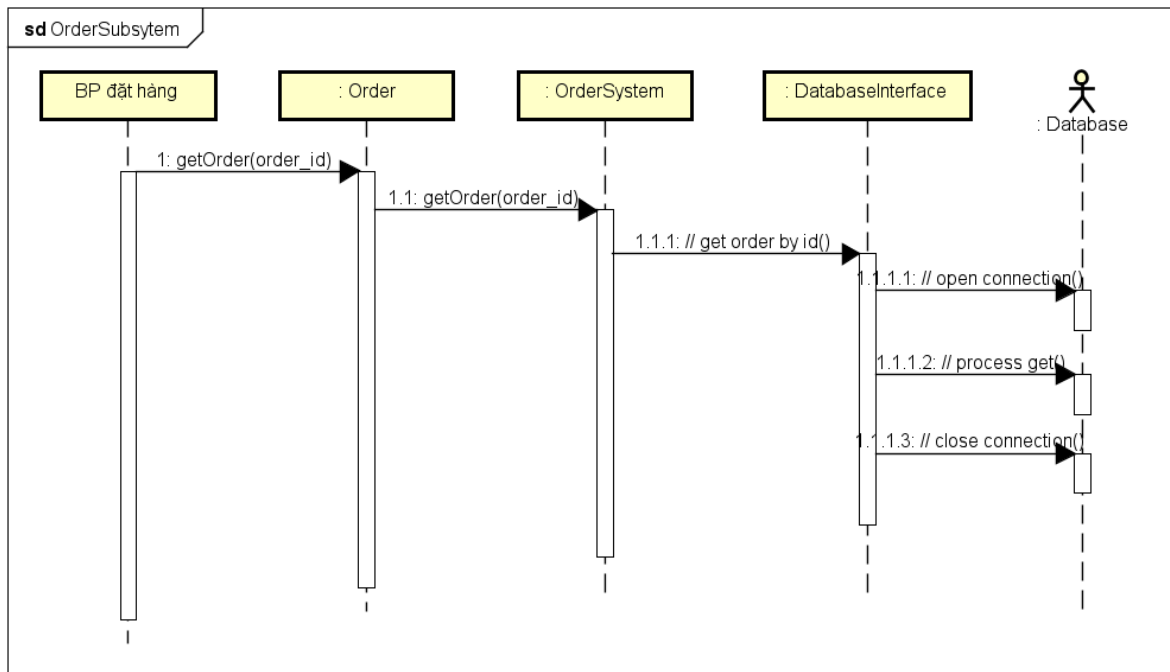
Note:

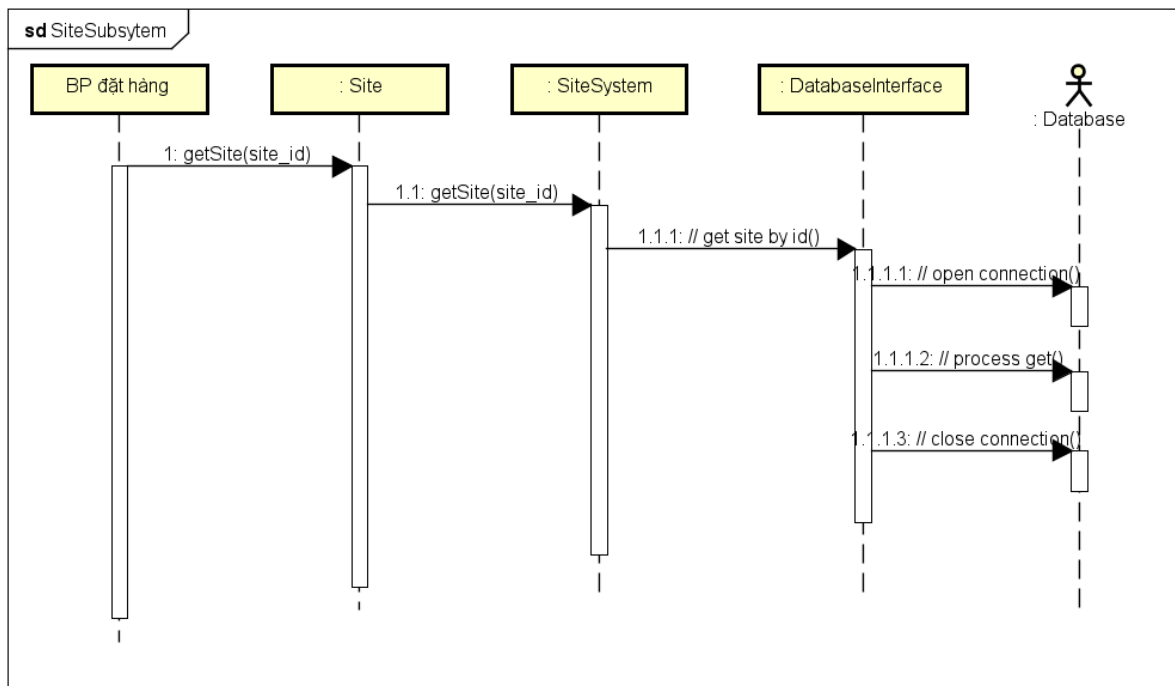
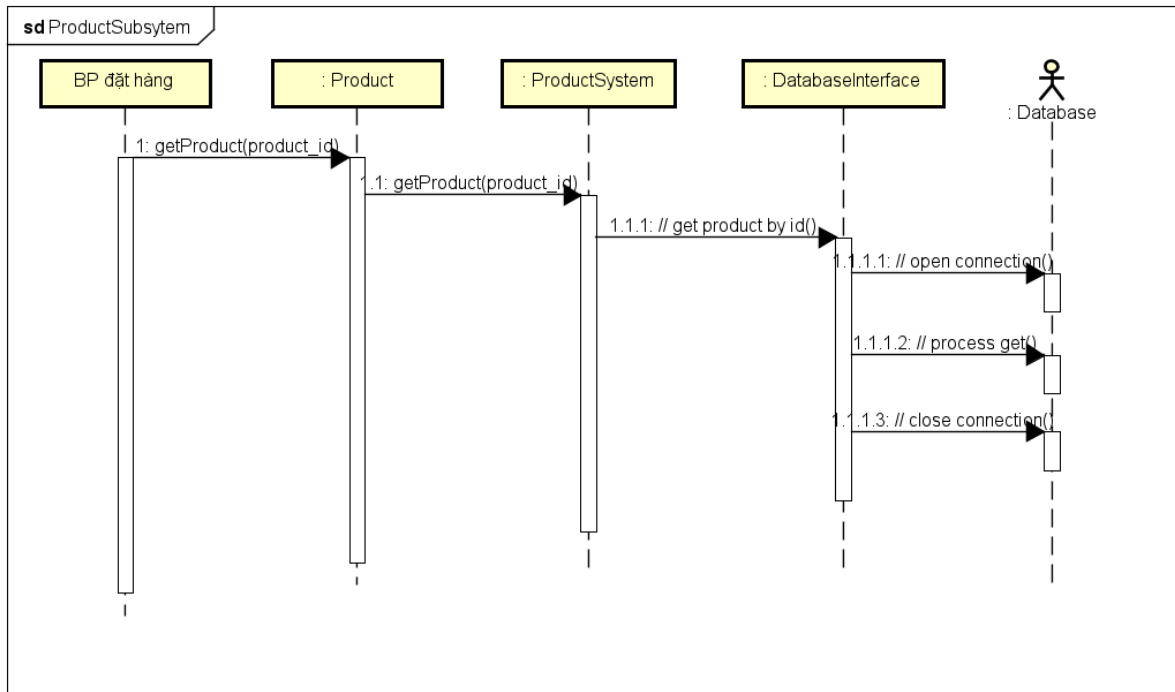
Quay Lại

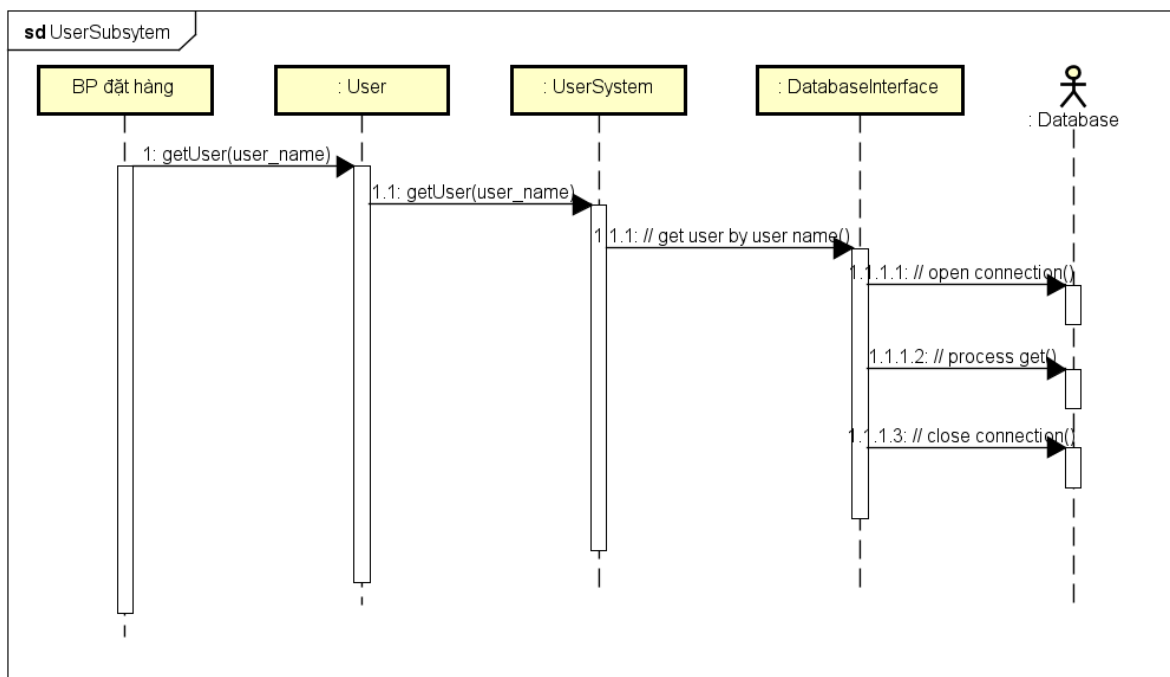
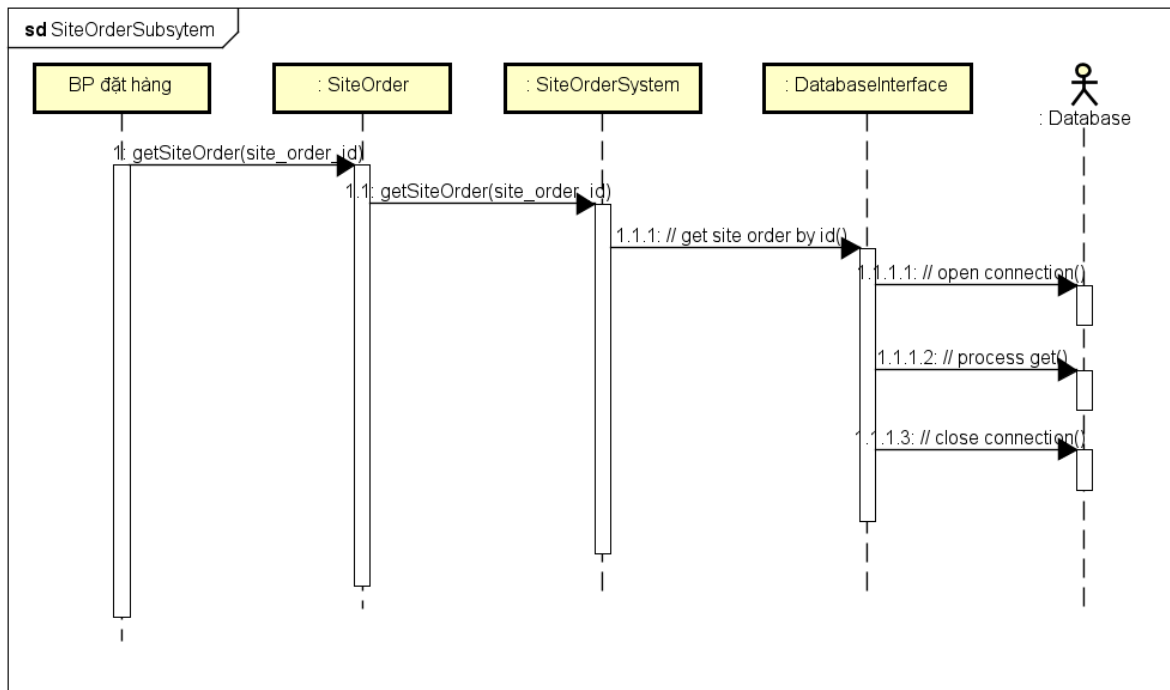
Nhập Kho

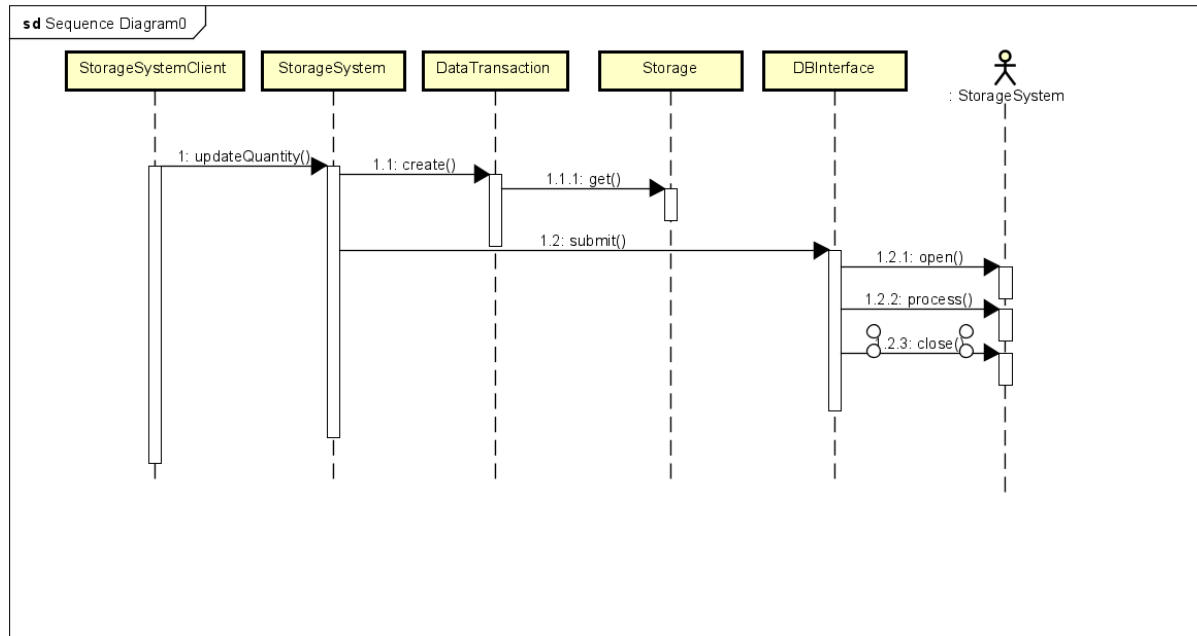
Và 1 số pop up khác.

- Subsystem:





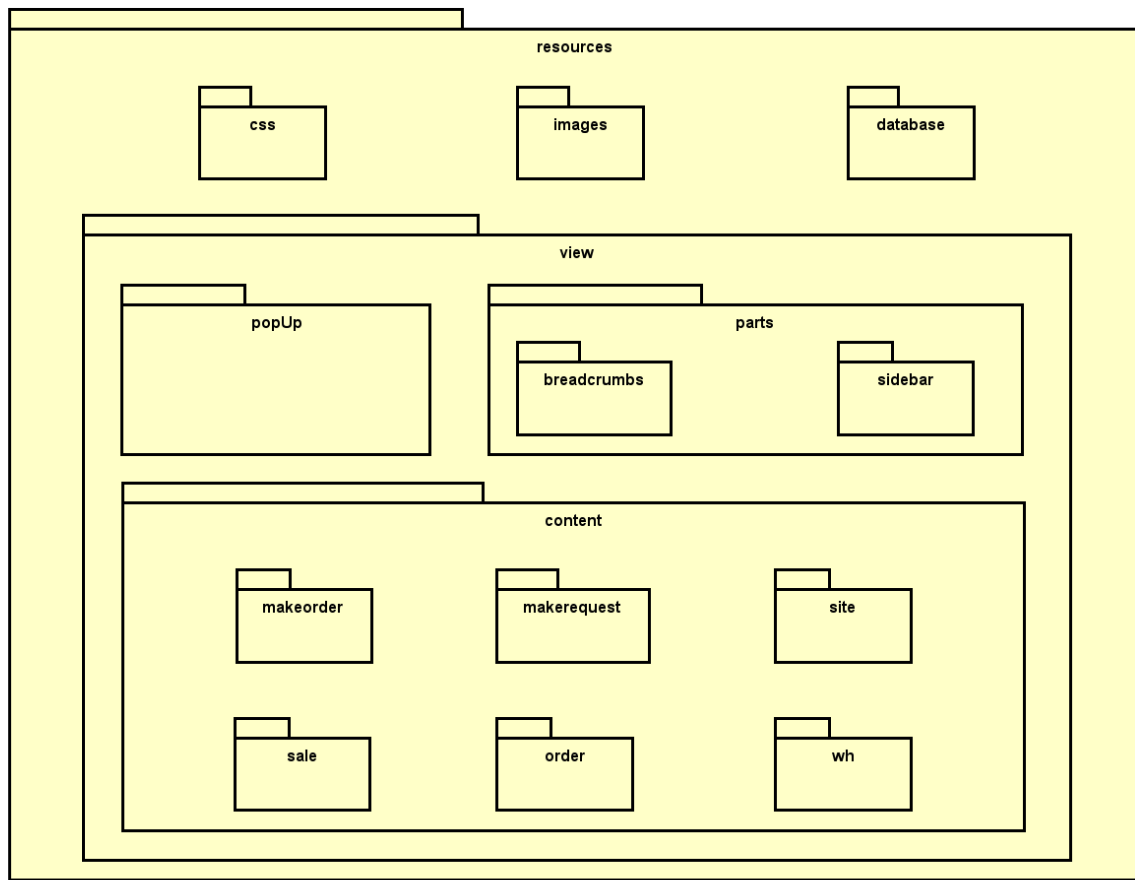
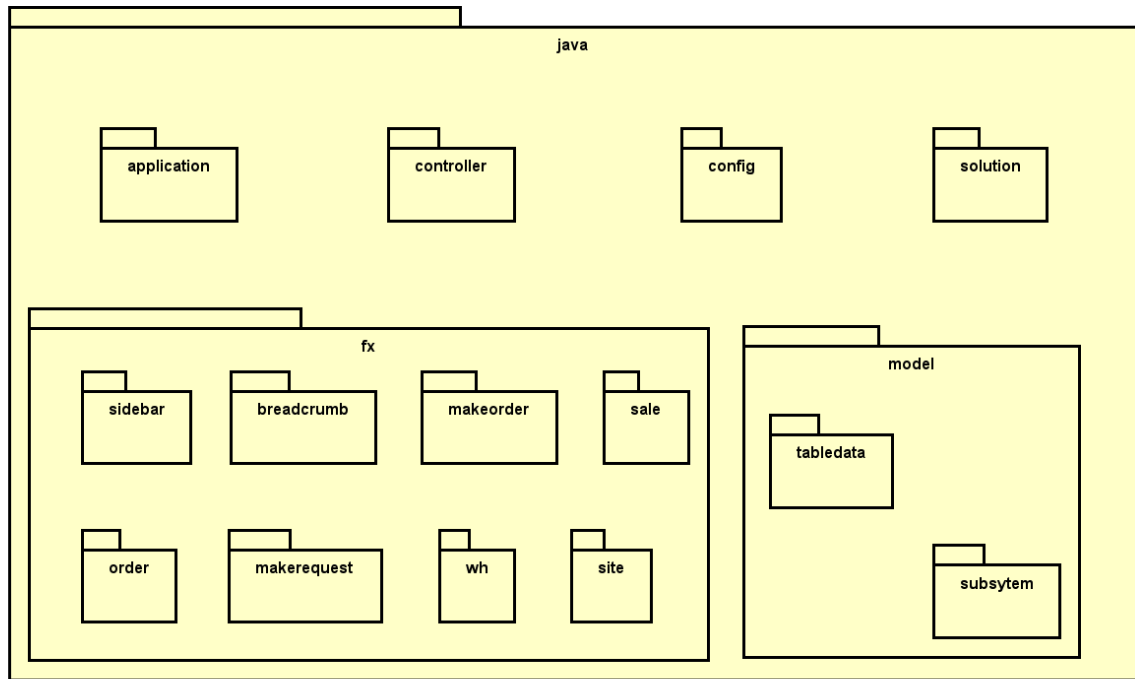


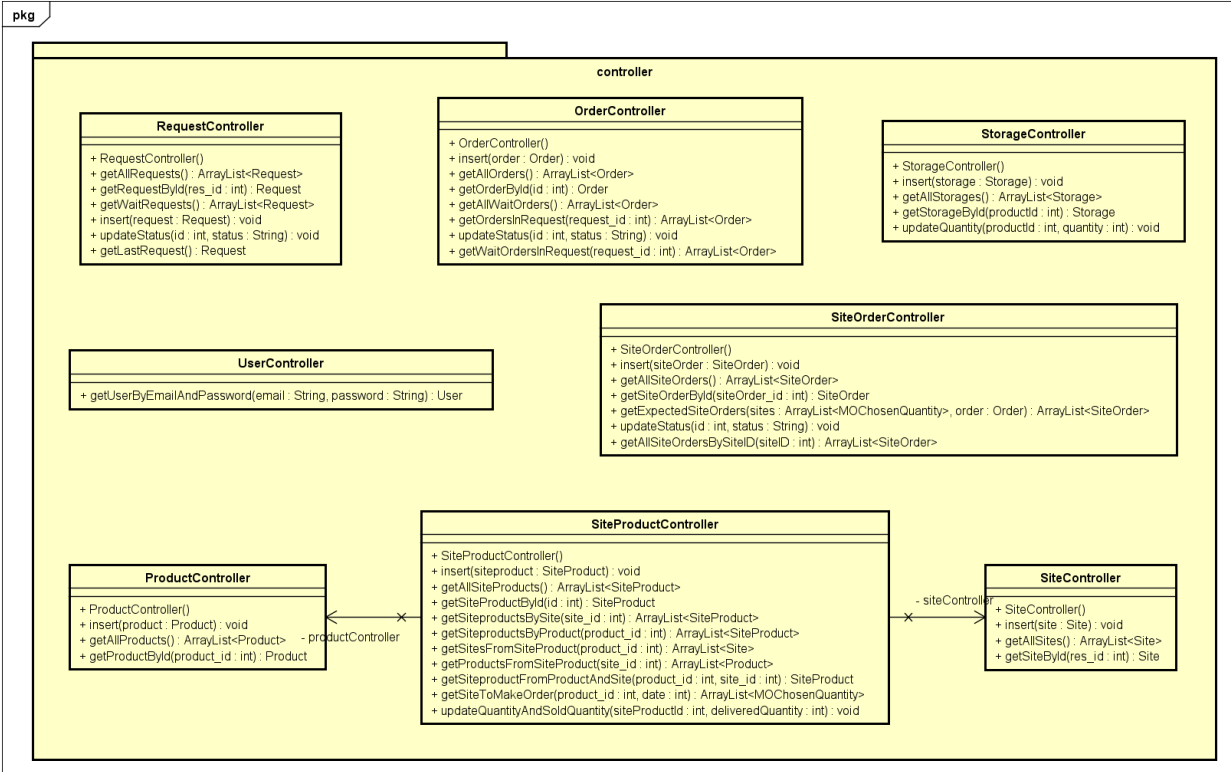


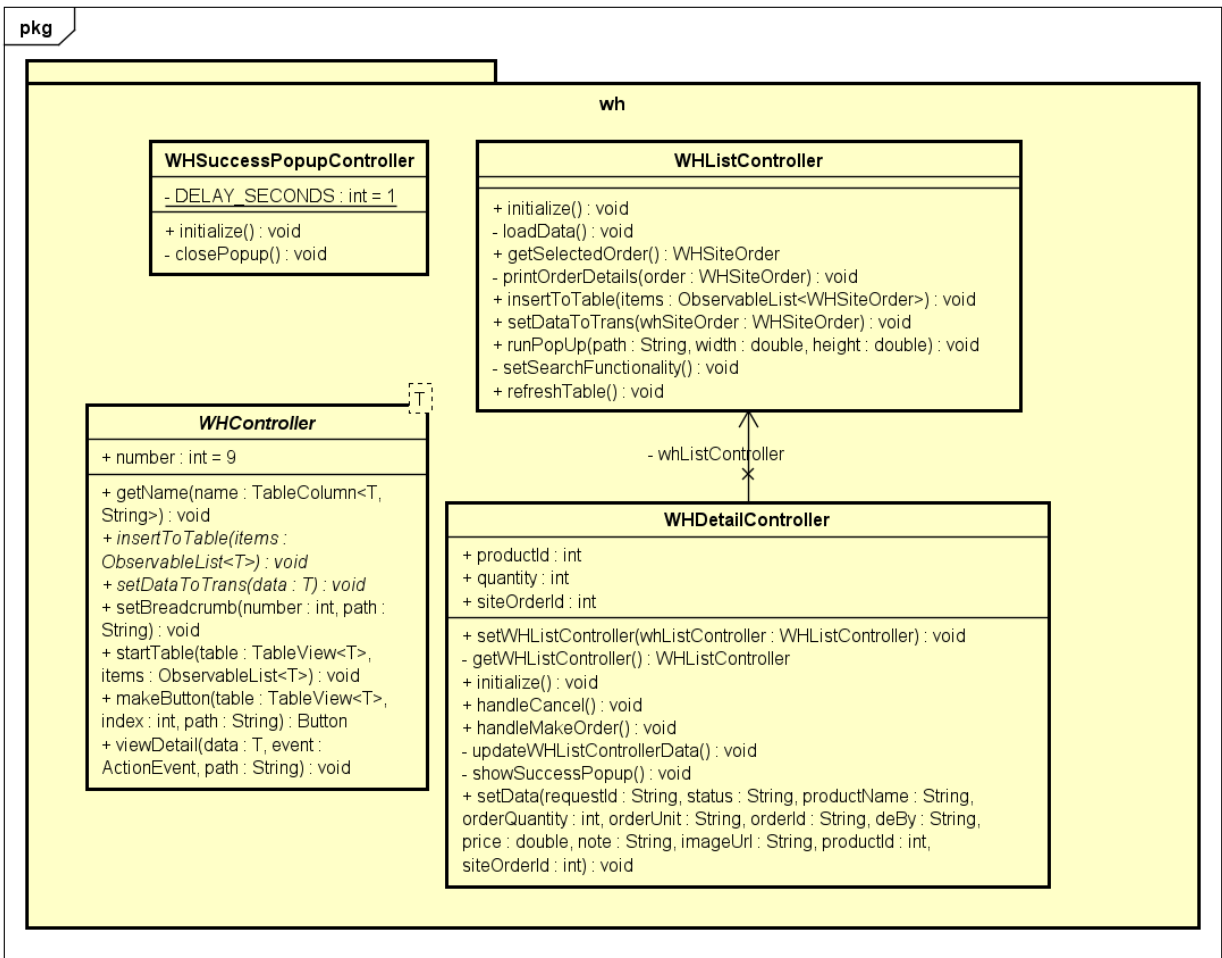
Bài 5

- Biểu đồ lớp

pkg







pkg

tabledata

WHSiteOrder

- idCounter : int = 1
- id : int
- maDonHang : String
- site : String
- phuongThuc : String
- trangThai : String
- note : String
- price : double

+ WHSiteOrder(siteOrder : SiteOrder, action : Button)
+ getId() : int
+ getSiteOrder() : SiteOrder
+ getMaDonHang() : String
+ getSite() : String
+ getPhuongThuc() : String
+ getTrangThai() : String
+ getNote() : String
+ getPrice() : double
+ getAction() : Button
+ resetIdCounter() : void

pkg

solution

Paginator

+ setPagination(table : TableView<T>, pagination :
Pagination, items : ObservableList<T>, number : int) :
void

T

subsystem

SiteSystem

```
+ insert(site : Site) : void
+ delete(id : int) : void
+ update(site : Site) : void
+ selectAll() : ArrayList<Site>
+ selectById(id : int) : Site
```

ProductSystem

```
+ insert(product : Product) : void
+ delete(id : int) : void
+ update(product : Product) : void
+ selectAll() : ArrayList<Product>
+ selectById(id : int) : Product
```

SiteOrderSystem

```
+ insert(siteOrder : SiteOrder) : void
+ delete(id : int) : void
+ update(siteOrder : SiteOrder) : void
+ selectAll() : ArrayList<SiteOrder>
+ selectById(id : int) : SiteOrder
+ updateStatus(id : int, status : String) : void
+ selectBySiteId(siteId : int) : ArrayList<SiteOrder>
```

StorageSystem

```
+ insert(storage : Storage) : void
+ delete(id : int) : void
+ update(storage : Storage) : void
+ selectAll() : ArrayList<Storage>
+ selectById(id : int) : Storage
+ updateQuantity(productId : int, quantity : int) : void
```

RequestSystem

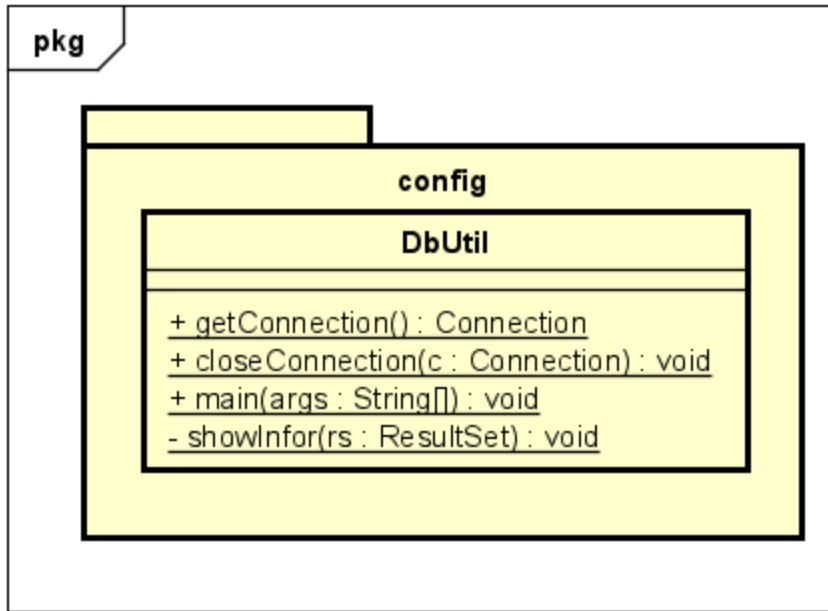
```
+ insert(request : Request) : void
+ update(request : Request) : void
+ delete(id : int) : void
+ selectById(id : int) : Request
+ selectAll() : ArrayList<Request>
+ selectByStatus(status : String) : ArrayList<Request>
+ updateStatus(id : int, status : String) : void
+ selectLast() : Request
```

UserSystem

```
+ selectTypeByEmailAndPass(email : String, pass : String) : User
```

SiteProductSystem

```
+ insert(siteproduct : SiteProduct) : void
+ delete(id : int) : void
+ update(siteproduct : SiteProduct) : void
+ selectAll() : ArrayList<SiteProduct>
+ selectById(id : int) : SiteProduct
+ selectBySiteId(siteId : int) : ArrayList<SiteProduct>
+ selectByProductId(productId : int) : ArrayList<SiteProduct>
+ selectSiteByProductId(productId : int) : ArrayList<Site>
+ selectProductBySiteId(siteId : int) : ArrayList<Product>
+ selectByProductAndSite(productId : int, siteId : int) : SiteProduct
+ selectChosingSite(productId : int, date : int) : ArrayList<MOChosenQuantity>
+ update2(siteproduct : SiteProduct) : void
```



pkg

fx

LoginController

+ sidebarPath : String
+ contentPath : String
+ imagePath : String

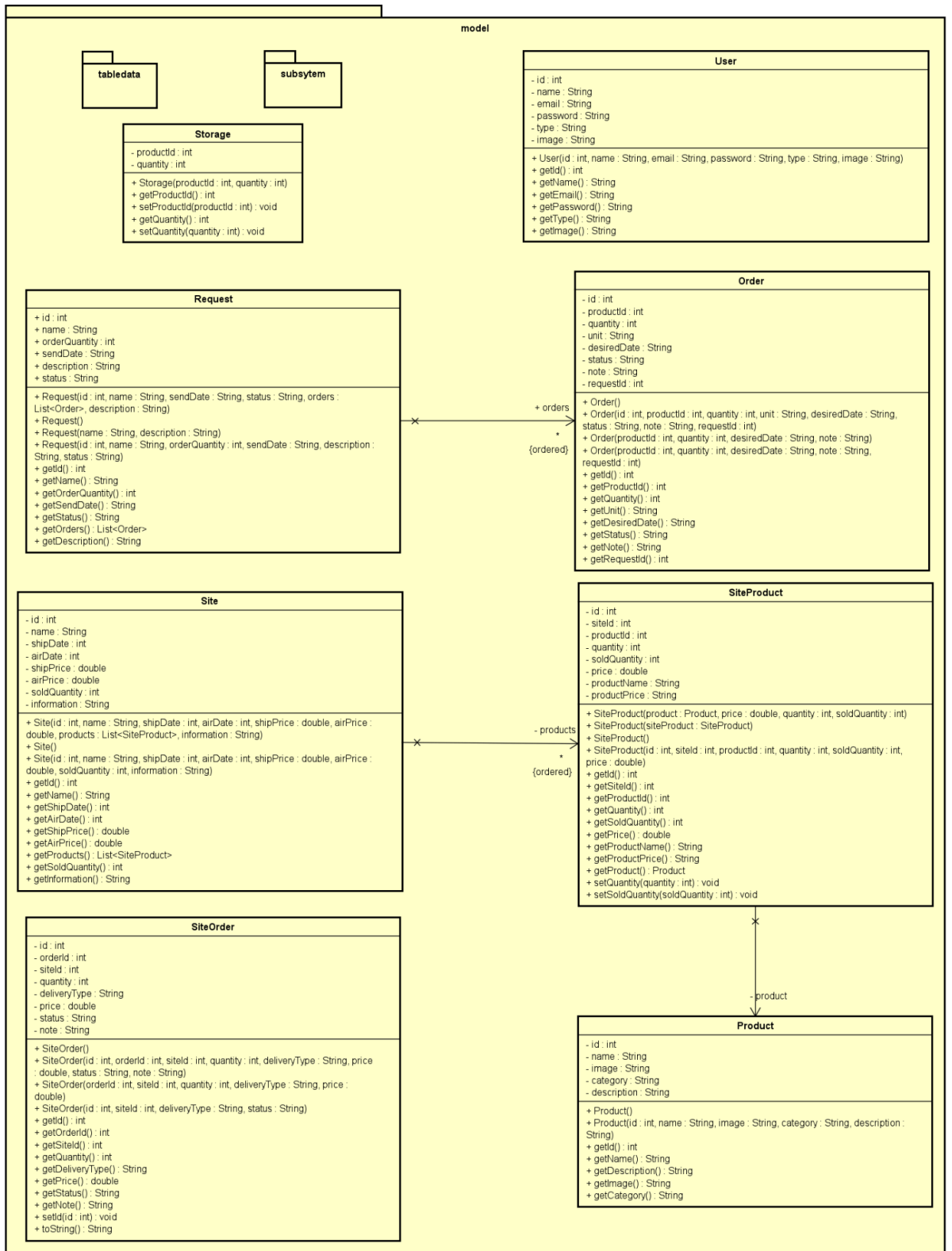
+ initialize() : void
- login() : void
- login(sb : String, ct : String, img : String) : void

MainController

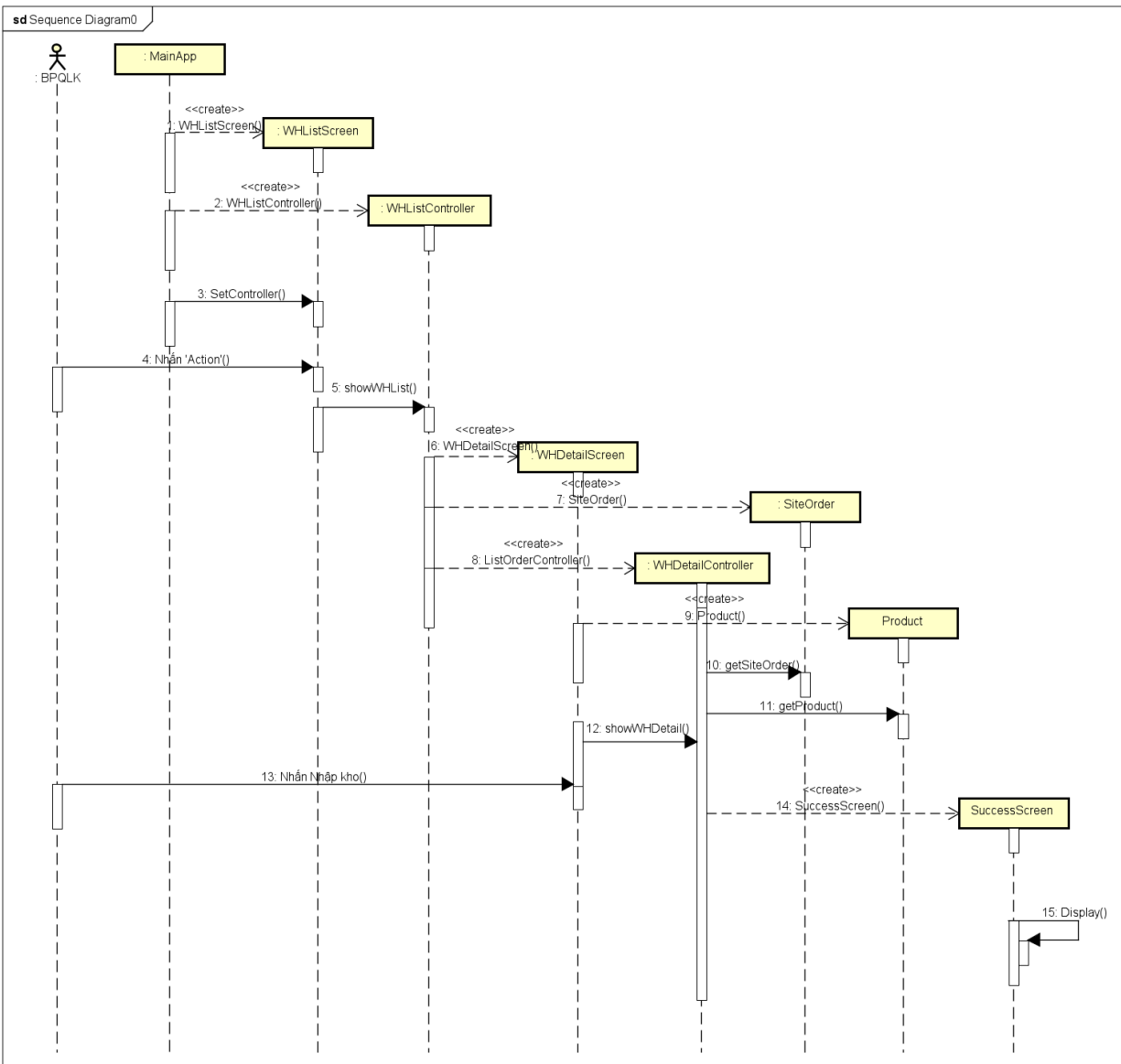
+ avatarPath : String
+ sidebarPath : String
+ contentPath : String
- avatarStatus : boolean = false

+ MainController()
+ setSidebarPath(sidebarPath : String) : void
+ setContentPath(contentPath : String) : void
+ setAvatarPath(avatarPath : String) : void
+ initialize() : void
- loadAvatar(path : String) : void
- loadSidebar(path : String) : void
- loadContent(path : String) : void
- clickAvatar() : void
~ logout(event : ActionEvent) : void

pkg



• Biểu đồ trình tự mức thiết kế:



Bài 6

- Thiết kế hệ thống con (subsystem): StorageSystem là một subsystem chịu trách nhiệm quản lý và thao tác với dữ liệu liên quan đến kho hàng trong cơ sở dữ liệu. Nó cung cấp các phương thức để thêm, xóa, cập nhật và truy xuất thông tin về kho hàng. StorageSystem tuân thủ giao diện DBInterface<Storage>
- Thiết kế cấu trúc project: gom nhóm các lớp có chức năng liên quan chung 1 package, khi cần sửa 1 chức năng thì tìm kiếm lớp đó dễ dàng hơn.
- Nguyên lý SOLID

- Phân chia module backend và frontend thành 2 modules tách biệt không ảnh hưởng tới nhau (nguyên tắc số 5)
- Mỗi đối tượng sẽ có một controllers, service để xử lý riêng (nguyên tắc số 1)

Bài 7

Kiểm thử lớp “ProductController”

❖ Kiểm thử hộp đen

+ Phương thức getProductById:

- Đầu vào: ProductID
- Dự kiến đầu ra: Sản phẩm với ID tương ứng.

Testcase:

- Lấy sản phẩm bằng 1 ID hợp lệ
- Lấy sản phẩm bằng 1 ID không hợp lệ

Kỹ thuật kiểm thử phân vùng tương đương: những ID có trong database sẽ hợp lệ và ngược lại.

❖ Chương trình kiểm thử tự động: ProductControllerTest

The screenshot shows an IDE with the following components:

- Project Explorer:** Displays a project structure with packages like `StorageSystem`, `UserSystem`, and `tabledata`. The `ProductControllerTest` class is highlighted under the `controller` package.
- Editor:** Shows the source code of `ProductControllerTest.java`. The code includes:


```
package controller;

import ...

public class ProductControllerTest {
    private ProductController productController;

    @BeforeEach
    public void setUp() { productController = new ProductController(); }

    @Test
    public void testGetAllProducts() {
        ArrayList<Product> products = productController.getAllProducts();
        // Ensure that the list of products is not null and not empty
        assertNotNull(products);
        assertFalse(products.isEmpty());
    }

    @Test
    public void testGetProductById() {
```
- Run Console:** Shows the execution of the tests. It indicates that 2 tests passed out of 2 tests, with a total time of 367 ms. The tests are `testGetProductById()` (360 ms) and `testGetAllProducts()` (7 ms). The process finished with exit code 0.

