

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

-----00-----



SOICT

BÁO CÁO HỌC PHẦN THIẾT KẾ VÀ XÂY DỰNG PHẦN MỀM

*Đề tài: Phân tích, thiết kế và xây dựng
Phần mềm thương mại điện tử AIMS*

Giảng viên hướng dẫn: TS. Nguyễn Thị Thu Trang

Mã lớp học: 144930

Sinh viên thực hiện: Nhóm TKXDPM.KHMT.20231-09

Nguyễn Chí Hiếu	20200226
Nguyễn Cao Bảo Hiếu	20200225
Nguyễn Đình Hiếu	20204550
Phạm Minh Hiếu	20204553

HÀ NỘI, NGÀY 04 THÁNG 01 NĂM 2023

MỤC LỤC

1. Thông tin thành viên nhóm 09	4
2. Tổng hợp công việc của các thành viên trong nhóm	4
3. Phân tích, thiết kế và xây dựng Phần mềm thương mại điện tử AIMS	5
3.1. Phần chung	5
1. Phân tích hệ thống.....	5
2. Thiết kế hệ thống.....	6
3.2. Phần cá nhân thực hiện	30
3.2.1. Use case Quản lý đơn đặt hàng (Nguyễn Chí Hiếu).....	30
3.2.2. Use case Hủy đơn hàng (Nguyễn Cao Bảo Hiếu)	38
3.2.3. Use case Sắp xếp sản phẩm và Tìm kiếm sản phẩm (Nguyễn Đình Hiếu)	43
3.2.4. Use case CRUD sản phẩm (Phạm Minh Hiếu)	48

1. Thông tin thành viên nhóm 09

Họ và tên	Vai trò
Nguyễn Chí Hiếu	Nhóm trưởng
Nguyễn Cao Bảo Hiếu	Thành viên
Nguyễn Đình Hiếu	Thành viên
Phạm Minh Hiếu	Thành viên

2. Tổng hợp công việc của các thành viên trong nhóm

2.1. Nguyễn Chí Hiếu

- Dựng base code Capstone Project và phân công nhiệm vụ cho các thành viên trong nhóm
- Tìm hiểu và tìm lỗi Content Coupling, Common Coupling trong base code
- Tìm temporal cohesion và procedural cohesion trong base code
- Tìm và sửa lỗi trong base code nếu vi phạm nguyên tắc Single Responsibility Principle
- Thực hiện phân tích, thiết kế và xây dựng use case “Quản lý đơn đặt hàng” (bao gồm phân tích, thiết kế và xây dựng)
- Thực hiện phần Data modeling: Er diagram, database design(logical data model, physical data model và SQL code)
- Tổng hợp và viết báo cáo
- Vẽ các biểu đồ chung liên quan đến project (Usecase tổng quan, General Class Diagram, Interface Design)

2.2. Nguyễn Cao Bảo Hiếu

- Thực hiện phân tích, thiết kế và xây dựng use case “Hủy đơn hàng” (bao gồm phân tích, thiết kế và xây dựng)
- Tìm hiểu và tìm lỗi Data Coupling trong base code
- Tìm coincidental cohesion và logical cohesion trong base code
- Tìm và sửa lỗi trong base code nếu vi phạm nguyên tắc Liskov Substitution và Interface Segregation Principle
- Quay video demo Project
- Tổng hợp đặc tả chi tiết use case của các thành viên
- Viết báo cáo

2.3. Nguyễn Đình Hiếu

- Thực hiện phân tích, thiết kế và xây dựng use case “Sắp xếp sản phẩm theo giá”, “Tìm kiếm sản phẩm” (bao gồm phân tích, thiết kế và xây dựng)
- Tìm hiểu và tìm lỗi Control Coupling trong base code
- Tìm communicational cohesion và functional cohesion trong base code
- Tìm và sửa lỗi trong base code nếu vi phạm nguyên tắc Open Closed principle
- Làm slide thuyết trình
- Viết báo cáo

2.4. Phạm Minh Hiếu

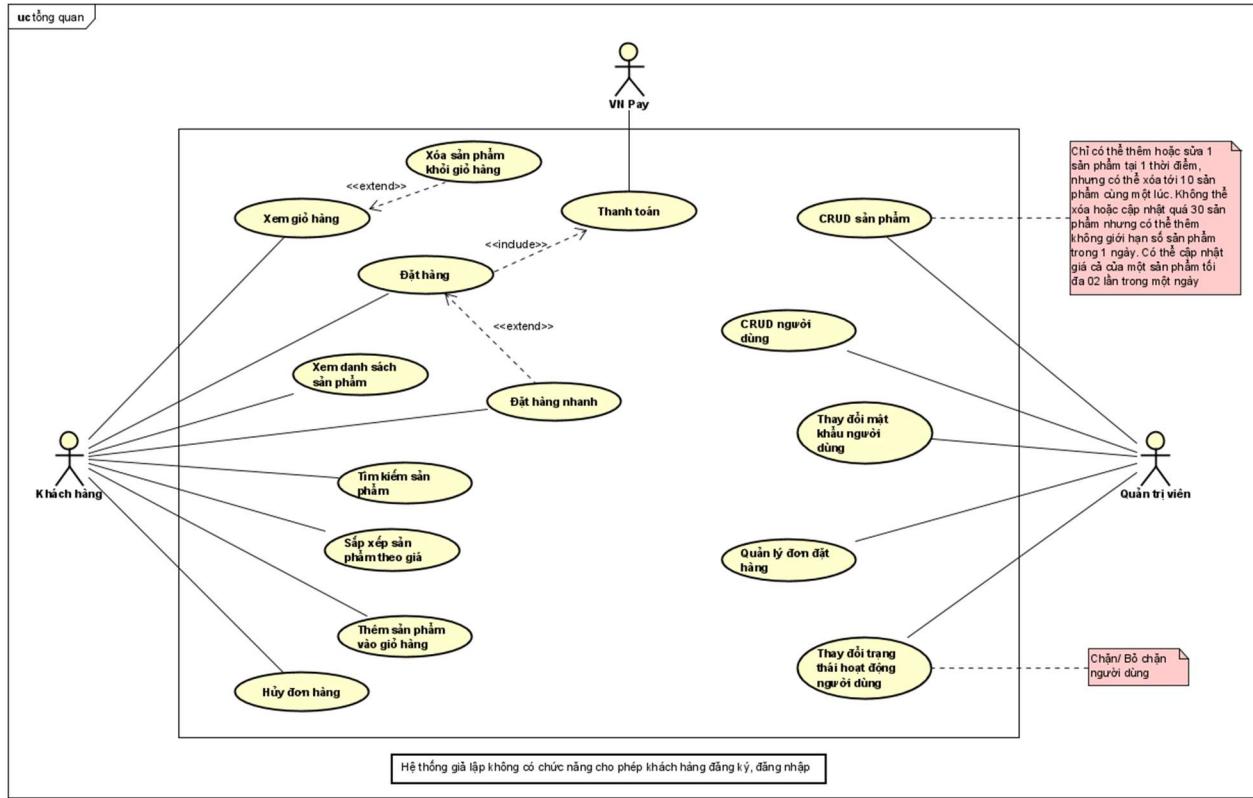
- Thực hiện phân tích, thiết kế và xây dựng use case “CRUD sản phẩm” (bao gồm phân tích, thiết kế và xây dựng)
- Tìm hiểu và tìm lỗi Stamp Coupling trong base code
- Kiểm tra lại xem các cohesion các thành viên khác tìm được đã hợp lý chưa
- Tìm và sửa lỗi trong base code nếu vi phạm nguyên tắc Dependency Inversion Principle
- Vẽ biểu đồ Relationship class diagram
- Viết báo cáo

3. Phân tích, thiết kế và xây dựng Phần mềm thương mại điện tử AIMS

3.1. Phần chung

1. Phân tích hệ thống

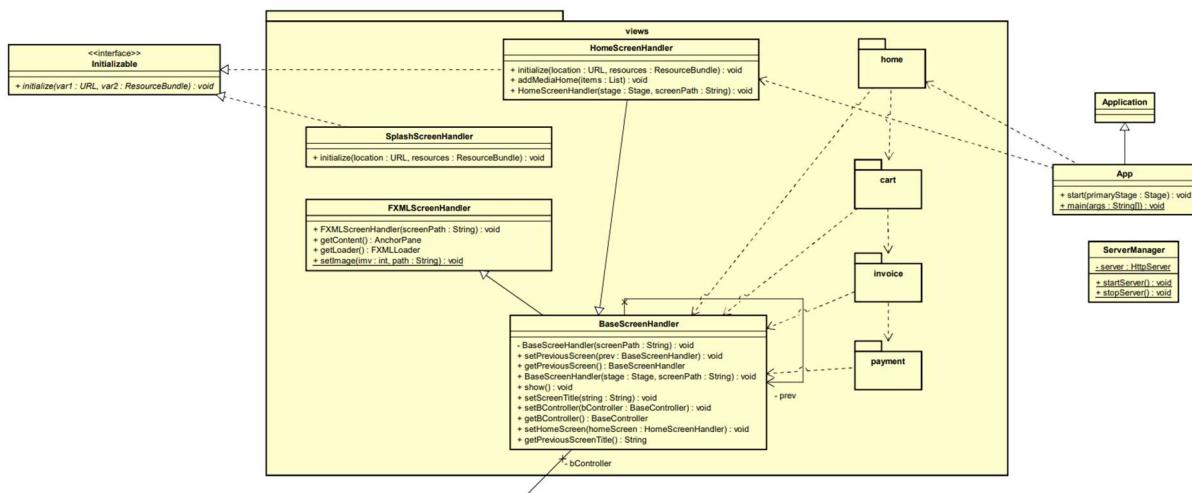
- Use case tổng quan



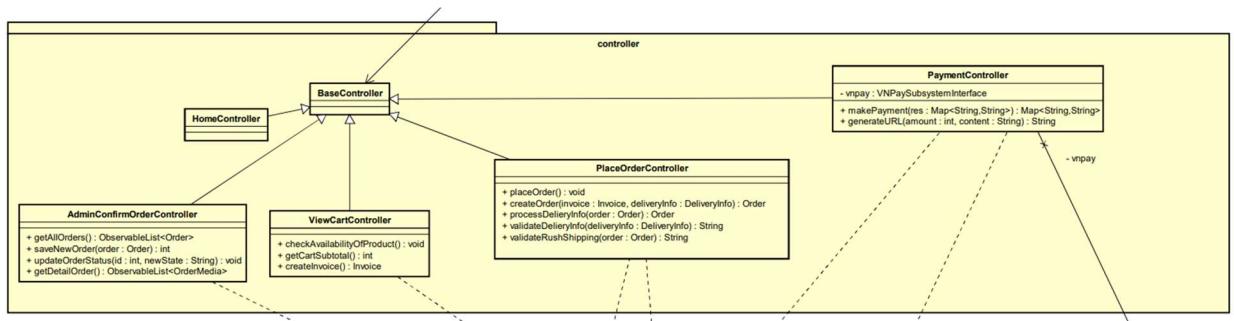
2. Thiết kế hệ thống

2.1. General Class Diagram

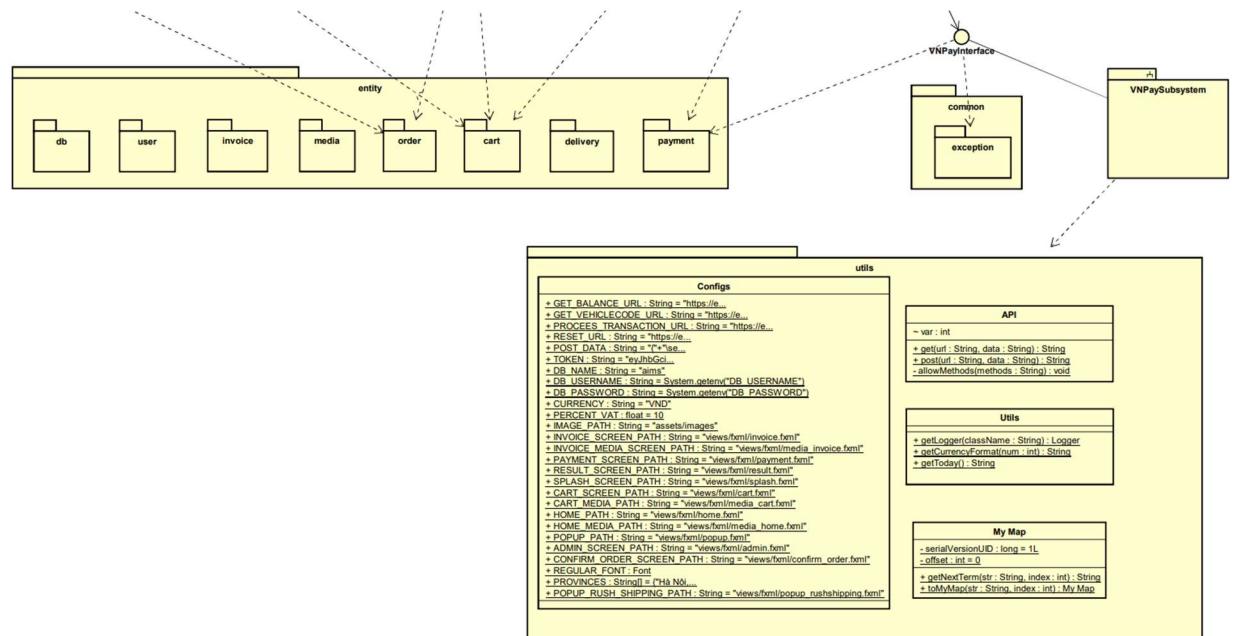
- views



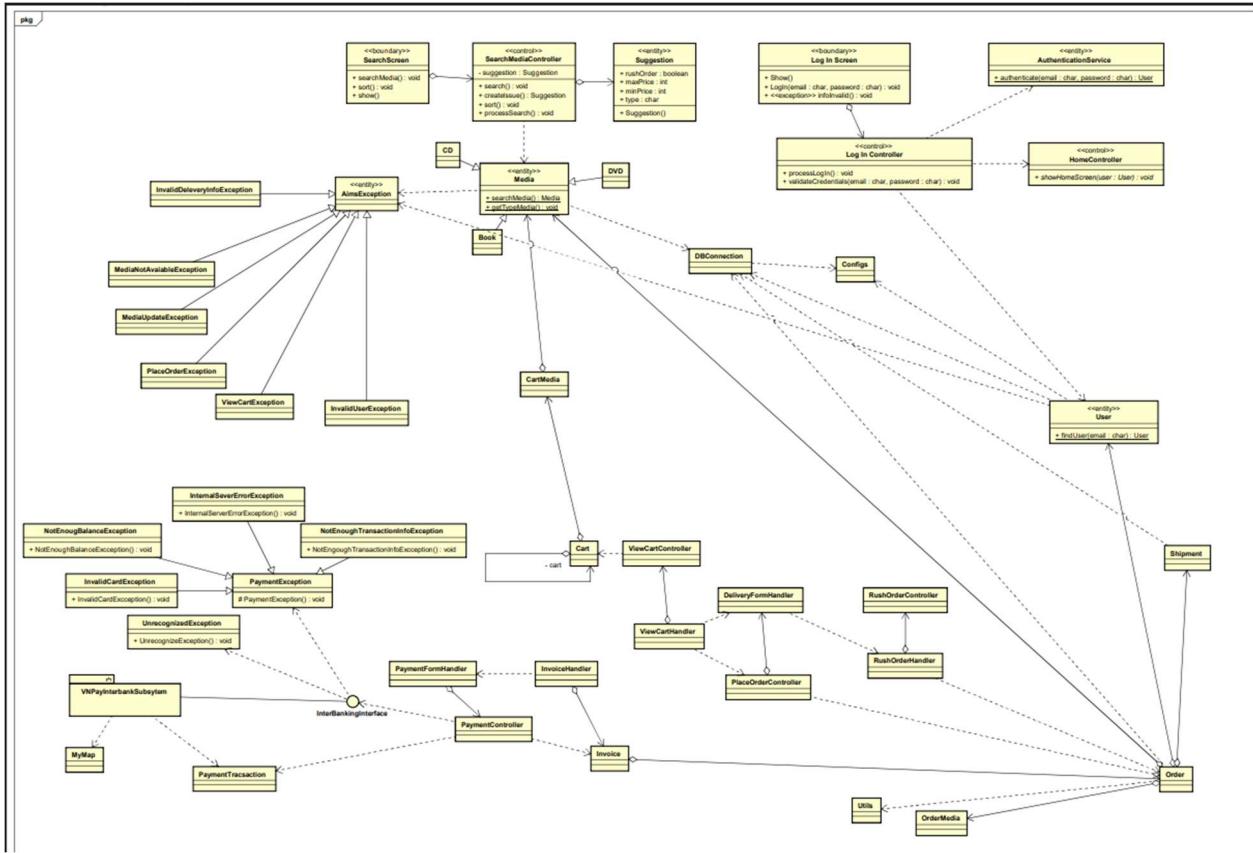
- controller



- entity

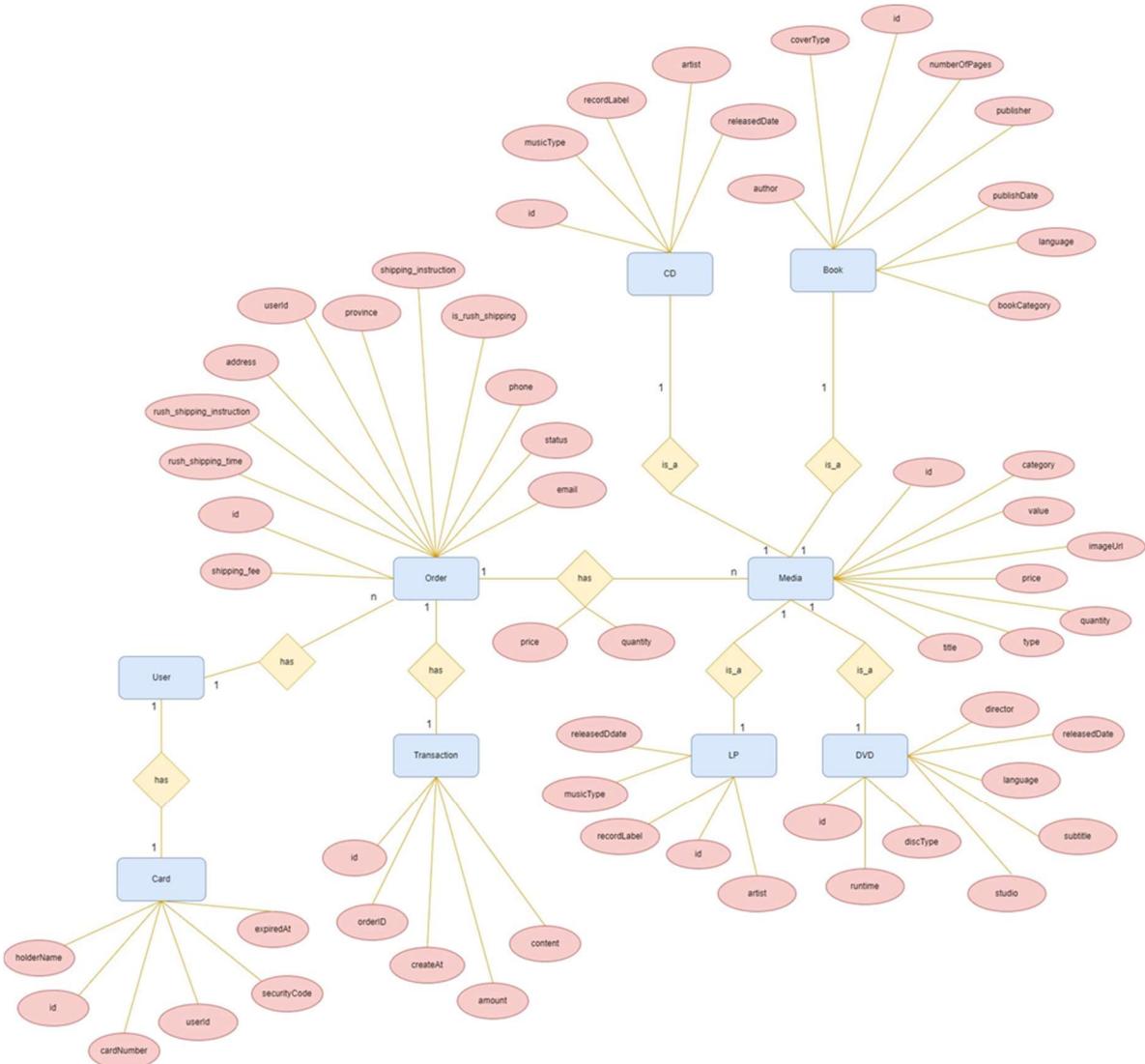


2.2. Relationship Class Diagram

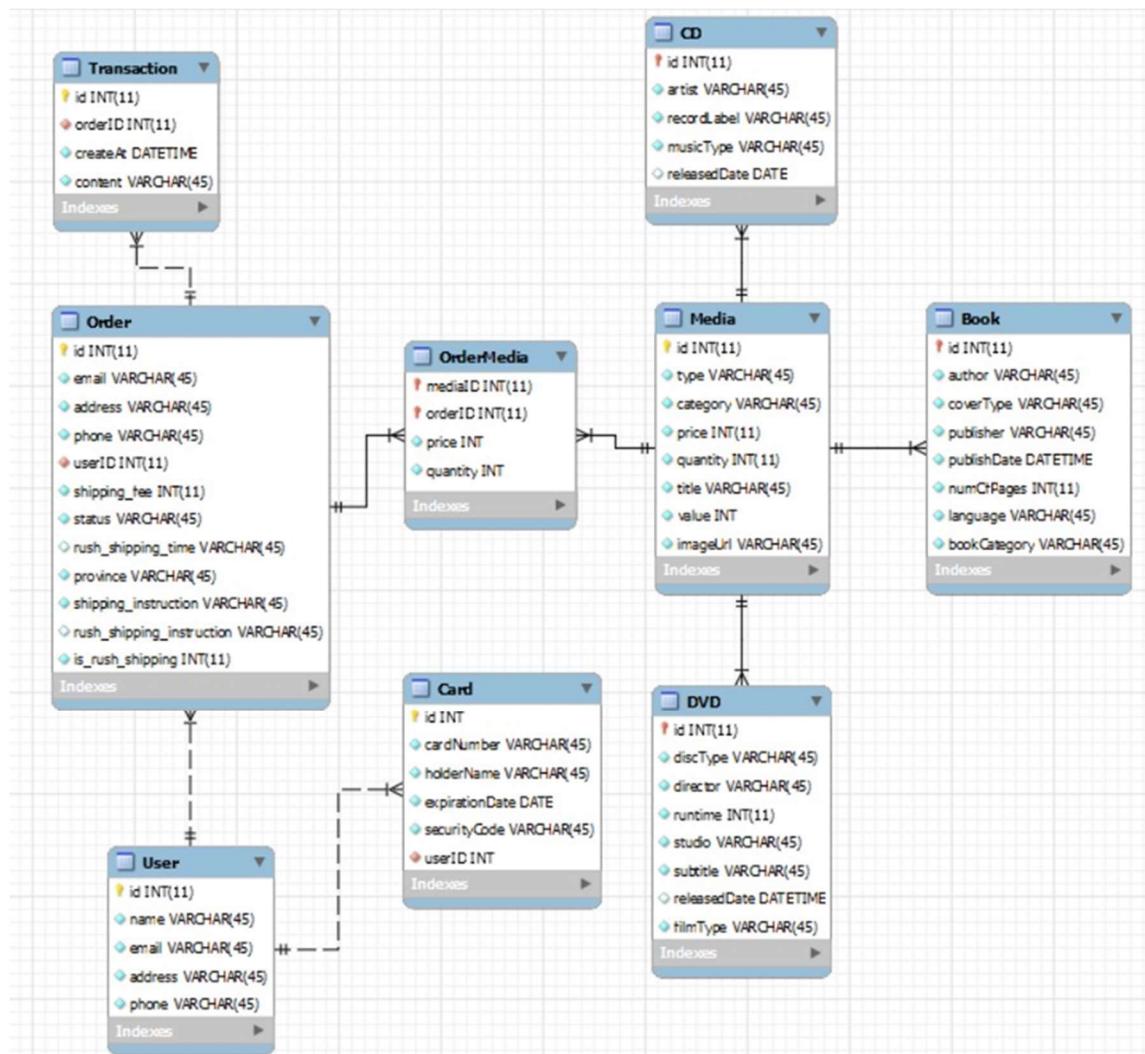


2.2. Data Modeling

1. Conceptual data model: Entity-Relation Diagram (ERD)



2. Logical data model: Database Design Diagram



3. Physical data model

3.1. Table User

STT	PK	FK	Column Name	Data Type	Mandatory	Description
1	x		id	INT	Yes	ID, auto increment
2			name	VARCHAR(45)	Yes	User name
3			email	VARCHAR(45)	Yes	User email
4			address	VARCHAR(45)	Yes	User address
5			phone	VARCHAR(45)	Yes	User contact number

3.2. Table Media

STT	PK	FK	Column Name	Data Type	Mandatory	Description
1	x		id	INT	Yes	ID, auto increment
2			price	INT	Yes	Current price
3			category	VARCHAR(45)	Yes	Media category (story, pop, adventure)
4			value	INT	Yes	Product value

5			quantity	INT	Yes	Number of products
6			title	VARCHAR(45)	Yes	Product name
7			imageUrl	VARCHAR(45)	No	Product image path
8			type	VARCHAR(45)	Yes	Media type (CD, DVD, Book, LP)

3.3. Table CD

STT	PK	FK	Column Name	Data Type	Mandatory	Description
1	x	x	id	INT	Yes	ID, same ID of Media of which type is CD
2			artist	VARCHAR(45)	Yes	Artist's name
3			recordLabel	VARCHAR(45)	Yes	Record Label
4			musicType	VARCHAR(45)	Yes	Music genres
6			releasedDate	DATETIME	No	Release Date

3.4. Table Book

STT	PK	FK	Column Name	Data Type	Mandatory	Description
1	x	x	id	INT	Yes	ID, same ID of Media of which type is Book
2			author	VARCHAR(45)	Yes	Author's name
3			coverType	VARCHAR(45)	Yes	Cover Type
4			numberOfPages	INT	Yes	Page number
5			language	VARCHAR(45)	Yes	Language
6			bookCategory	VARCHAR(45)	Yes	Type of Book
7			publisher	VARCHAR(45)	Yes	Publishing house
8			publishDate	DATETIME	Yes	Date of publishing

3.5. Table DVD

STT	PK	FK	Column Name	Data Type	Mandatory	Description

1	x	x	id	INT	Yes	ID, same ID of Media of which type is DVD
2			director	VARCHAR(45)	Yes	Director's name
3			discType	VARCHAR(45)	Yes	Disc Type
4			subtitle	VARCHAR(45)	Yes	Subtitles
5			studio	VARCHAR(45)	Yes	Manufacturer
6			language	VARCHAR(45)	Yes	Language
7			runtime	INT	Yes	Duration
8			releasedDate	DATETIME	No	Release Date

3.6. Table LP

STT	PK	FK	Column Name	Data Type	Mandatory	Description
1	x	x	id	INT	Yes	ID, same ID of Media of which type is LP
2			artist	VARCHAR(45)	Yes	Artist's name
3			recordLabel	VARCHAR(45)	Yes	Record Label
4			musicType	VARCHAR(45)	Yes	Music genres

6			releasedDate	DATETIME	No	Release Date
----------	--	--	--------------	----------	----	--------------

1.1. Table Order

STT	PK	FK	Column Name	Data Type	Mandatory	Description
1	x		id	INT	Yes	ID, auto increment
2			email	VARCHAR(45)	Yes	User's email
3			address	VARCHAR(45)	Yes	User's address
4			phone	VARCHAR(45)	Yes	User's phone number
5		x	userID	INT	Yes	User ID
6			shipping_fee	INT	Yes	Shipping fee
7			status	VARCHAR(45)	Yes	Order status(Chờ duyệt, Đã duyệt, Đã từ chối)

8			rush_shipping_time	VARCHAR(45))	No	Rush shipping time
9			province	VARCHAR(45))	Yes	Province
10			shipping_instruction	VARCHAR(45))	Yes	Instruction of shipping
11			rush_shpping_instruction	VARCHAR(45))	No	Instruction of rush shipping
12			is_rush_shipping	INT	Yes	Shipping method

2.1. Table OrderMedia

STT	PK	FK	Column Name	Data Type	Mandatory	Description
2	x	x	mediaId	INT	Yes	Media ID
3	x	x	orderId	INT	Yes	Order ID
4			price	INT	Yes	Selling price

5			quantity	INT	Yes	Number of ordered item
---	--	--	----------	-----	-----	------------------------

3.1. Table Transaction

STT	PK	FK	Column Name	Data Type	Mandatory	Description
1	x		id	INT	Yes	ID, auto increment
3			content	VARCHAR(45)	Yes	Transaction contents
4			amount	INT	Yes	Payment Amount
5			createAt	DATETIME	Yes	Date of creation a transaction
7		x	orderID	INT	Yes	ID of order

4.1. Table Card

STT	PK	FK	Column Name	Data Type	Mandatory	Description
1	x		id	INT	Yes	ID, auto increment

2			cardNumber	VARCHAR(45)	Yes	Card Number of customer
3			holderName	VARCHAR(45)	Yes	Card holders
4		x	userId	VARCHAR(45)	Yes	User ID
5			securityCode	VARCHAR(45)	Yes	Bank Code
6			expirationDate	DATETIME	Yes	Expire Date

- **Database Script from Logical Data Model**
- BEGIN TRANSACTION;
- CREATE TABLE IF NOT EXISTS "Media" (

```

"id" INTEGER NOT NULL,
"type"VARCHAR(45) NOT NULL,
"category" VARCHAR(45) NOT NULL,
"price"      INTEGER NOT NULL,
"quantity"   INTEGER NOT NULL,
"title"      VARCHAR(45) NOT NULL,
"value"      INTEGER NOT NULL,
"imageUrl"   VARCHAR(45) NOT NULL,
PRIMARY KEY("id" AUTOINCREMENT)
);
```

- CREATE TABLE IF NOT EXISTS "CD" (
- "id" INTEGER NOT NULL,

```
    "artist"      VARCHAR(45) NOT NULL,  
  
    "recordLabel"     VARCHAR(45) NOT NULL,  
  
    "musicType" VARCHAR(45) NOT NULL,  
  
    "releasedDate"   DATE,  
  
    CONSTRAINT "fk_cd_media" FOREIGN KEY("id") REFERENCES  
    "Media"("id"),  
  
    PRIMARY KEY("id")  
  
);
```

- CREATE TABLE IF NOT EXISTS "Book" (

```
    "id"      INTEGER NOT NULL,  
  
    "author"     VARCHAR(45) NOT NULL,  
  
    "coverType" VARCHAR(45) NOT NULL,  
  
    "publisher"  VARCHAR(45) NOT NULL,  
  
    "publishDate" DATETIME NOT NULL,  
  
    "numOfPages"   INTEGER NOT NULL,  
  
    "language"    VARCHAR(45) NOT NULL,  
  
    "bookCategory"  VARCHAR(45) NOT NULL,  
  
    CONSTRAINT "fk_book_media" FOREIGN KEY("id") REFERENCES  
    "Media"("id"),  
  
    PRIMARY KEY("id" AUTOINCREMENT)  
  
);
```

- CREATE TABLE IF NOT EXISTS "User" (

```
    "id"      INTEGER NOT NULL,  
  
    "name"VARCHAR(45) NOT NULL,  
  
    "email"    VARCHAR(45) NOT NULL,
```

```

    "address"      VARCHAR(45) NOT NULL,
    "phone"        VARCHAR(45) NOT NULL,
    PRIMARY KEY("id" AUTOINCREMENT)
);

• CREATE TABLE IF NOT EXISTS "DVD" (
    "id"          INTEGER NOT NULL,
    "discType"    VARCHAR(45) NOT NULL,
    "director"    VARCHAR(45) NOT NULL,
    "runtime"     INTEGER NOT NULL,
    "studio"      VARCHAR(45) NOT NULL,
    "subtitle"    VARCHAR(45) NOT NULL,
    "releasedDate" DATETIME,
    "filmType"    VARCHAR(45) NOT NULL,
    CONSTRAINT "fk_dvd_media" FOREIGN KEY("id") REFERENCES
    "Media"("id"),
    PRIMARY KEY("id")
);

• CREATE TABLE IF NOT EXISTS "OrderMedia" (
    "mediaID"     INTEGER NOT NULL,
    "orderID"     INTEGER NOT NULL,
    "price"       INTEGER NOT NULL,
    "quantity"    INTEGER NOT NULL,
    CONSTRAINT "fk_ordermedia_media" FOREIGN KEY("mediaID") REFERENCES
    "Media"("id"),
    CONSTRAINT "fk_ordermedia_order" FOREIGN KEY("orderID") REFERENCES
    "Order"("id"),

```

```

PRIMARY KEY("mediaID","orderID")

);

• CREATE TABLE IF NOT EXISTS "Transaction" (
    "id" INTEGER NOT NULL,
    "orderID" INTEGER NOT NULL,
    "createAt" DATETIME NOT NULL,
    "content" VARCHAR(45) NOT NULL,
    CONSTRAINT "fk_transaction_order" FOREIGN KEY("orderID")
REFERENCES "Order"("id"),
    PRIMARY KEY("id" AUTOINCREMENT)

);

• CREATE TABLE IF NOT EXISTS "Card" (
    "id" INTEGER NOT NULL,
    "cardNumber" VARCHAR(45) NOT NULL,
    "holderName" VARCHAR(45) NOT NULL,
    "expirationDate" DATE NOT NULL,
    "securityCode" VARCHAR(45) NOT NULL,
    "userID" INTEGER NOT NULL,
    CONSTRAINT "fk_card_user" FOREIGN KEY("userID") REFERENCES
"User"("id"),
    PRIMARY KEY("id")

);

• CREATE TABLE IF NOT EXISTS "Order" (
    "id" INTEGER NOT NULL,
    "email" VARCHAR(45) NOT NULL,

```

```

"address"    VARCHAR(45) NOT NULL,
"phone"      VARCHAR(45) NOT NULL,
"userID"     INTEGER NOT NULL,
"shipping_fee"   INTEGER NOT NULL,
"status"      VARCHAR(45),
"rush_shipping_time"  VARCHAR(45),
"province"    VARCHAR(45) NOT NULL,
"shipping_instruction" VARCHAR(45),
"rush_shipping_instruction" VARCHAR(45),
"is_rush_shipping"   INTEGER,
CONSTRAINT "fk_order_user" FOREIGN KEY("userID") REFERENCES "User"
("id"),
PRIMARY KEY("id" AUTOINCREMENT)
);

```

- INSERT INTO "Media" VALUES
(38,'book','story',32,12,'book2',29,'assets/images/book/book2.jpg');
- INSERT INTO "Media" VALUES
(39,'book','adventure',21,2,'book9',20,'assets/images/book/book9.jpg');
- INSERT INTO "Media" VALUES
(40,'book','adventure',73,11,'book10',69,'assets/images/book/book10.jpg');
- INSERT INTO "Media" VALUES
(41,'book','story',66,2,'book6',62,'assets/images/book/book6.jpg');
- INSERT INTO "Media" VALUES
(42,'cd','pop',24,6,'cd7',20,'assets/images/cd/cd7.jpg');
- INSERT INTO "Media" VALUES
(43,'book','story',50,7,'book12',44,'assets/images/book/book12.jpg');
- INSERT INTO "Media" VALUES
(44,'book','story',57,10,'book4',53,'assets/images/book/book4.jpg');

- INSERT INTO "Media" VALUES
(45,'cd','pop',66,8,'cd3',60,'assets/images/cd/cd3.jpg');
- INSERT INTO "Media" VALUES
(46,'book','bussiness',79,17,'book1',72,'assets/images/book/book1.jpg');
- INSERT INTO "Media" VALUES
(47,'dvd','cartoon',82,1,'dvd12',78,'assets/images/dvd/dvd12.jpg');
- INSERT INTO "Media" VALUES
(48,'book','science',25,10,'book3',22,'assets/images/book/book3.jpg');
- INSERT INTO "Media" VALUES
(49,'dvd','science fiction',75,3,'dvd10',74,'assets/images/dvd/dvd10.jpg');
- INSERT INTO "Media" VALUES
(50,'book','bussiness',26,4,'book11',19,'assets/images/book/book11.jpg');
- INSERT INTO "Media" VALUES
(51,'dvd','action',61,18,'dvd11',52,'assets/images/dvd/dvd11.jpg');
- INSERT INTO "Media" VALUES
(52,'cd','rock',40,4,'cd4',35,'assets/images/cd/cd4.jpg');
- INSERT INTO "Media" VALUES
(53,'dvd','action',70,16,'dvd9',60,'assets/images/dvd/dvd9.jpg');
- INSERT INTO "Media" VALUES
(54,'dvd','romance',47,19,'dvd2',39,'assets/images/dvd/dvd2.jpg');
- INSERT INTO "Media" VALUES
(55,'cd','pop',74,6,'cd2',71,'assets/images/cd/cd2.jpg');
- INSERT INTO "Media" VALUES
(56,'cd','rock',70,20,'cd1',60,'assets/images/cd/cd1.jpg');
- INSERT INTO "Media" VALUES
(57,'book','adventure',38,2,'book8',36,'assets/images/book/book8.jpg');
- INSERT INTO "Media" VALUES
(58,'dvd','cartoon',55,13,'dvd3',51,'assets/images/dvd/dvd3.jpg');
- INSERT INTO "Media" VALUES
(59,'dvd','action',28,1,'dvd6',26,'assets/images/dvd/dvd6.jpg');
- INSERT INTO "Media" VALUES
(60,'dvd','romance',38,17,'dvd4',33,'assets/images/dvd/dvd4.jpg');

- INSERT INTO "Media" VALUES
(61,'cd','pop',42,15,'cd12',38,'assets/images/cd/cd12.jpg')
;
- INSERT INTO "Media" VALUES
(62,'book','bussiness',34,15,'book5',29,'assets/images/book/book5.jpg');
- INSERT INTO "Media" VALUES
(63,'cd','ballad',99,4,'cd5',92,'assets/images/cd/cd5.jpg')
;
- INSERT INTO "Media" VALUES
(64,'cd','pop',38,10,'cd8',32,'assets/images/cd/cd8.jpg');
- INSERT INTO "Media" VALUES
(65,'cd','classic',37,10,'cd6',31,'assets/images/cd/cd6.jpg')
;
- INSERT INTO "Media" VALUES
(66,'book','bussiness',93,15,'book7',88,'assets/images/book/book7.jpg');
- INSERT INTO "Media" VALUES
(67,'cd','classic',25,5,'cd9',23,'assets/images/cd/cd9.jpg')
;
- INSERT INTO "Media" VALUES
(68,'dvd','romance',71,4,'dvd5',64,'assets/images/dvd/dvd5.jpg');
- INSERT INTO "Media" VALUES
(69,'cd','pop',97,17,'cd10',89,'assets/images/cd/cd10.jpg')
;
- INSERT INTO "Media" VALUES
(70,'dvd','romance',47,19,'dvd8',37,'assets/images/dvd/dvd8.jpg');
- INSERT INTO "Media" VALUES
(71,'dvd','science fiction',95,11,'dvd1',92,'assets/images/dvd/dvd1.jpg');
- INSERT INTO "Media" VALUES
(72,'dvd','action',23,9,'dvd7',16,'assets/images/dvd/dvd7.jpg');
- INSERT INTO "Media" VALUES
(73,'cd','classic',28,3,'cd11',20,'assets/images/cd/cd11.jpg');
- CREATE INDEX IF NOT EXISTS
"OrderMedia.fk_ordermedia_order_idx" ON "OrderMedia" (
"orderID"
);

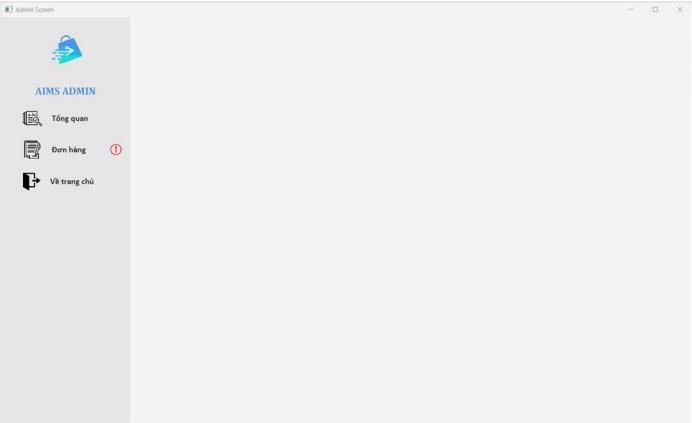
- CREATE INDEX IF NOT EXISTS "Transaction.fk_transaction_order_idx" ON "Transaction" (
 "orderID"
) ;
- CREATE INDEX IF NOT EXISTS "Card.fk_card_user_idx" ON "Card" (
 "userID"
) ;
- CREATE INDEX IF NOT EXISTS "Order.fk_order_user_idx" ON "Order" (
 "userID"
) ;
- COMMIT;

2.3. Graphical User Interface

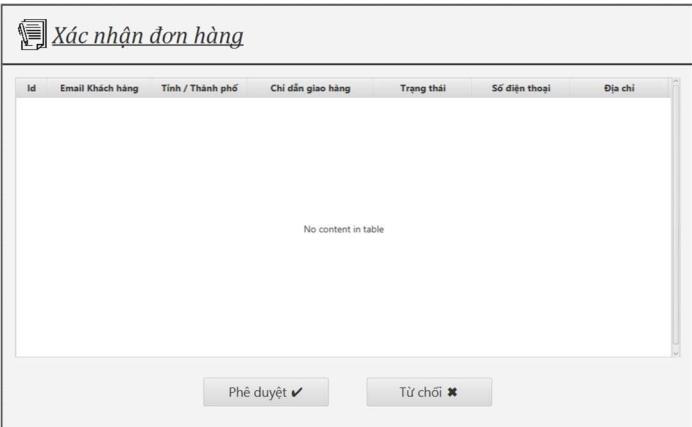
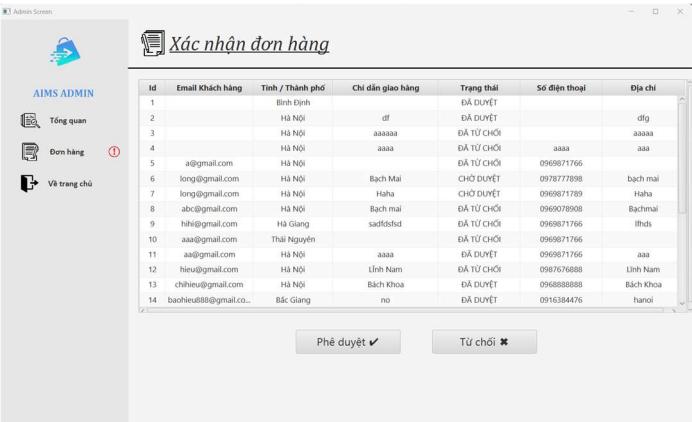
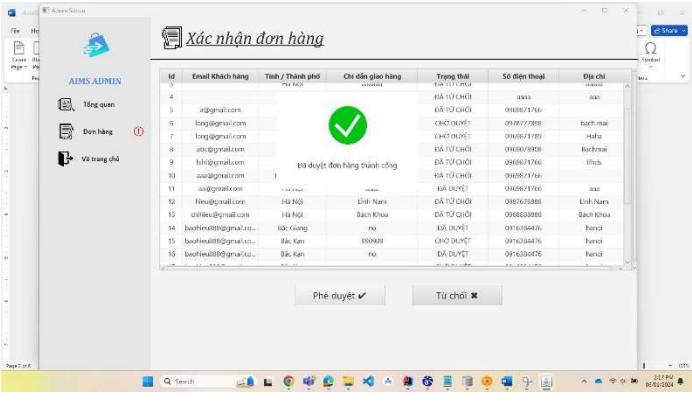
- Screen Specification

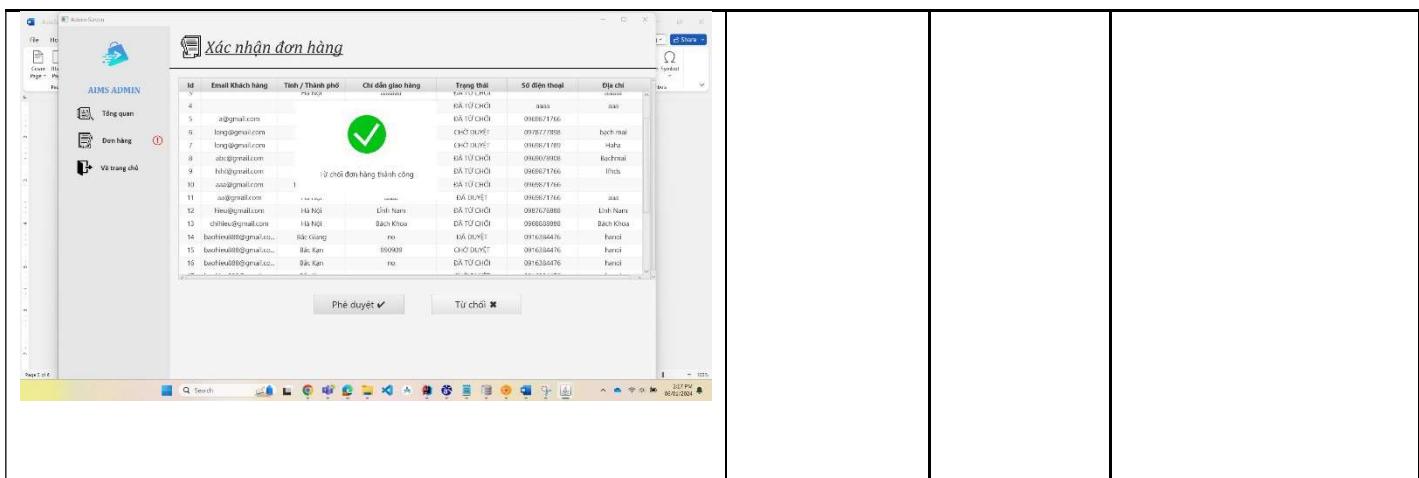
AIMS Software		Date of creation	Approved by	Reviewed by	Person in charge
Screen specification	Home Screen	15/12/2023			Nguyễn Chí Hiếu
		Control	Operation	Function	
		Khu vực hiển thị danh sách sản phẩm	Khởi tạo	Hiển thị danh sách sản phẩm của AIMS	

	Khu vực hiển thị ô tìm kiếm	Khởi tạo	Cho phép khách hàng nhập thông tin của sản phẩm cần tìm kiếm
Nút bấm “Tìm kiếm”	Nhấn chuột (Click)	Cho phép tìm kiếm theo trường vừa nhập	
Nút bấm “Sắp xếp theo giá”	Nhấn chuột (Click)	Cho phép sắp xếp các sản phẩm trên màn hình theo giá	
Trường nhập số lượng sản phẩm	Nhập dữ liệu	Cho phép khách hàng nhập số lượng sản phẩm thêm vào giỏ hàng	
Nút bấm tăng giảm số lượng	Click	Cho phép khách hàng tăng hoặc giảm số lượng sản phẩm thêm hoặc bớt 1 đơn vị	
Nút bấm “Thêm vào giỏ hàng”	Click	Cho phép khách hàng thêm sản phẩm với số lượng tương ứng vào giỏ hàng	
Nút bấm hình giỏ hàng	Click	Cho phép khách hàng chuyển sang màn xem giỏ hàng (Cart Screen)	
	Nút bấm hình Quản trị viên	Click	Cho phép quản trị viên chuyển sang màn dành cho Quản trị viên

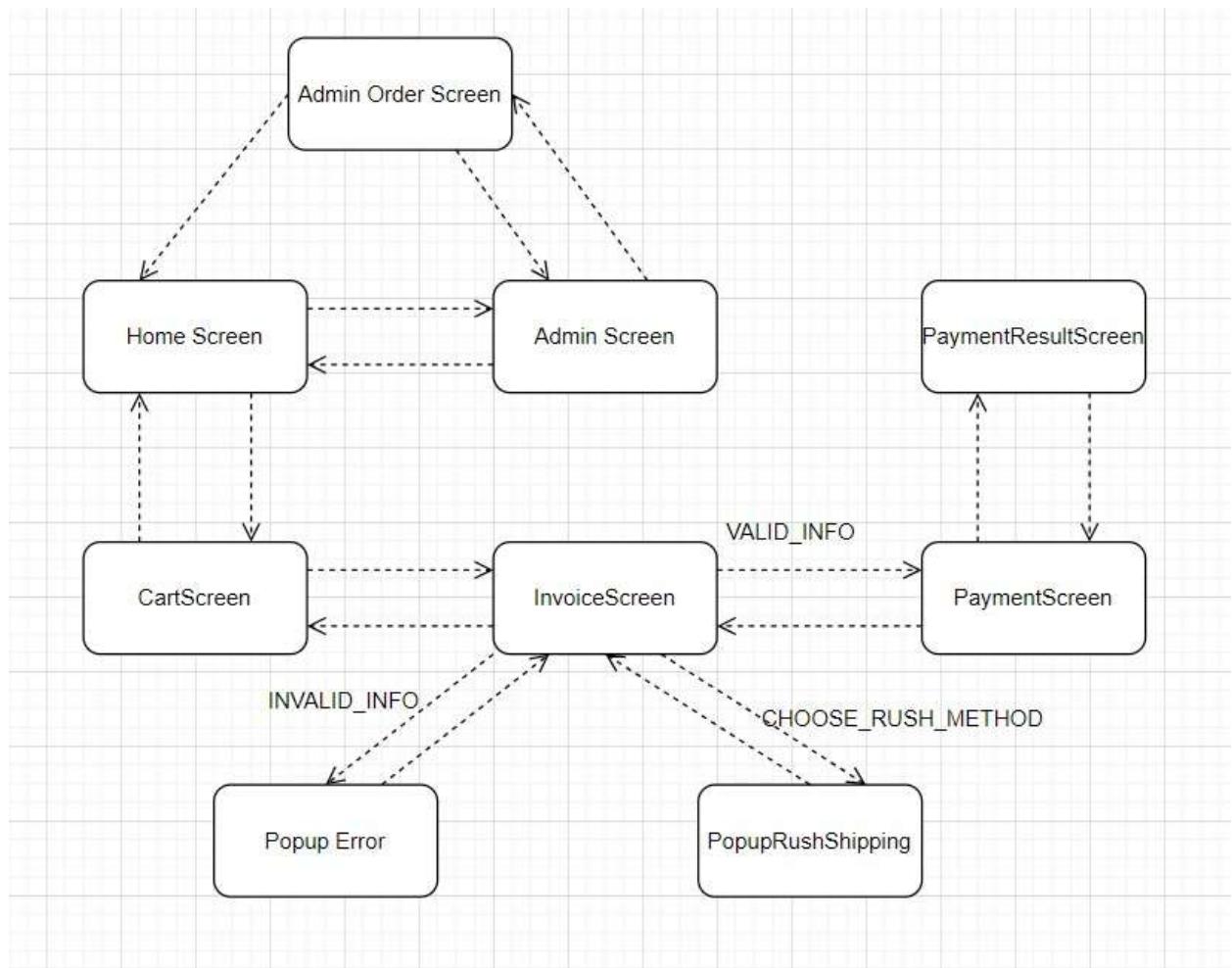
AIMS Software		Date of creation	Approved by	Reviewed by	Persion in charge
Screen specification	Admin Screen	20/12/2023			Nguyễn Chí Hiếu
		Control	Operation	Function	
		Khu vực Side bar hiển thị các chức năng trong màn Admin	Khởi tạo	Hiển thị các chức năng của hệ thống dành cho Quản trị viên	
	Nút bấm “Tổng quan”	Click	Hiển thị trang danh sách các sản phẩm trong hệ thống		
	Nút bấm “Đơn hàng”	Click	Hiển thị trang quản lý đơn hàng		
	Nút bấm “Về trang chủ”	Click	Cho phép Quay trở về trang chủ (ở cả Order Screen và Admin Screen)		
	Icon Logo AIMS	Click	Cho phép trở về trang chủ Home Screen		

AIMS Software		Date of creation	Approved by	Reviewed by	Persion in charge
Screen specification	Admin Order Screen	23/12/2023			Nguyễn Chí Hiếu
		Control	Operation	Function	

	Khu vực hiển thị bảng danh sách các đơn hàng đã đặt	Khởi tạo	Hiển thị popup thông báo cập nhật giỏ hàng
	Nút bấm “Phê duyệt”	Click	Cho phép quản trị viên cập nhật đơn hàng đang có trạng thái “Chờ duyệt” thành “Đã duyệt”
	Nút bấm “Từ chối”	Click	Cho phép quản trị viên cập nhật đơn hàng đang có trạng thái “Chờ duyệt” thành “Đã từ chối”
	Pop-up hiển thị cập nhật Đã duyệt / Đã từ chối đơn hàng thành công		Thông báo đến quản trị viên đã cập nhật thành công trạng thái đơn hàng



• Screen transition diagram

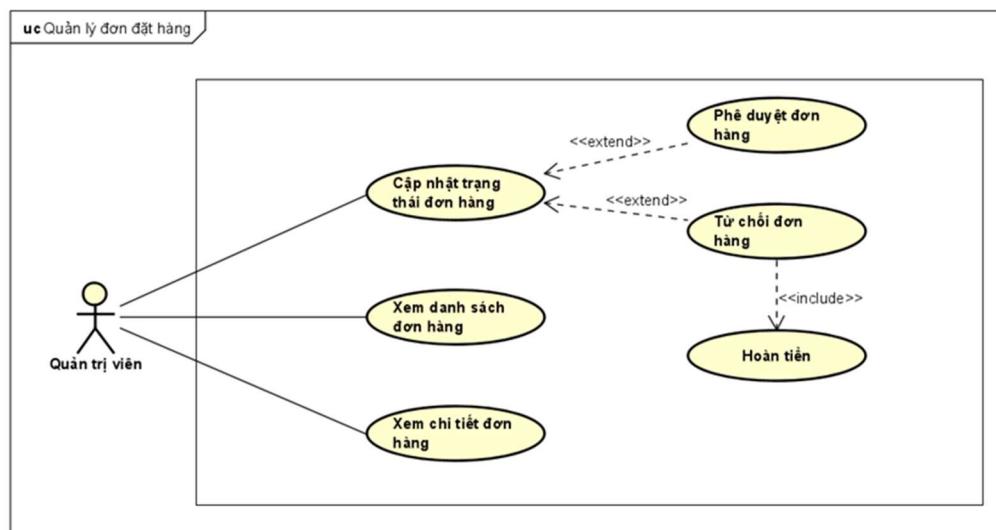


3.2. Phần cá nhân thực hiện

3.2.1. Use case Quản lý đơn đặt hàng (Nguyễn Chí Hiếu)

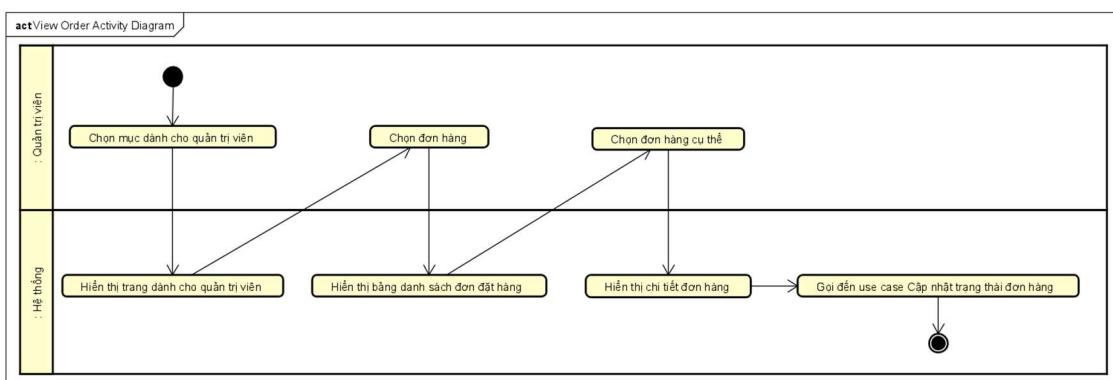
3.2.1.1. Phân tích use case Quản lý đơn đặt hàng

1. Phân rã use case Quản lý đơn đặt hàng

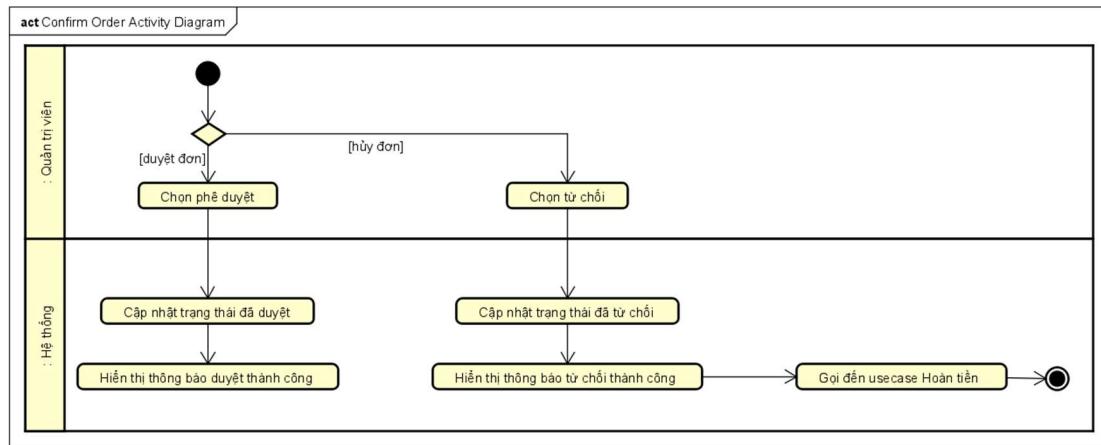


2. Activity Diagrams

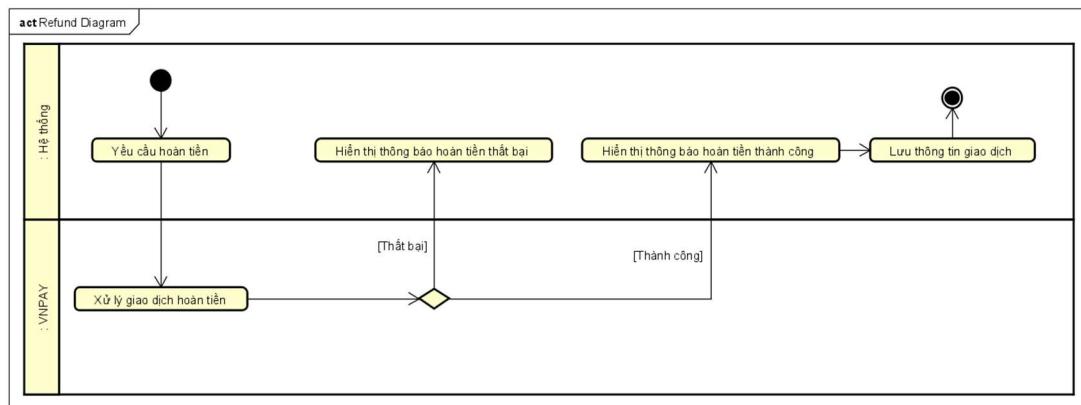
- Xem danh sách đơn hàng



- Cập nhật trạng thái đơn hàng

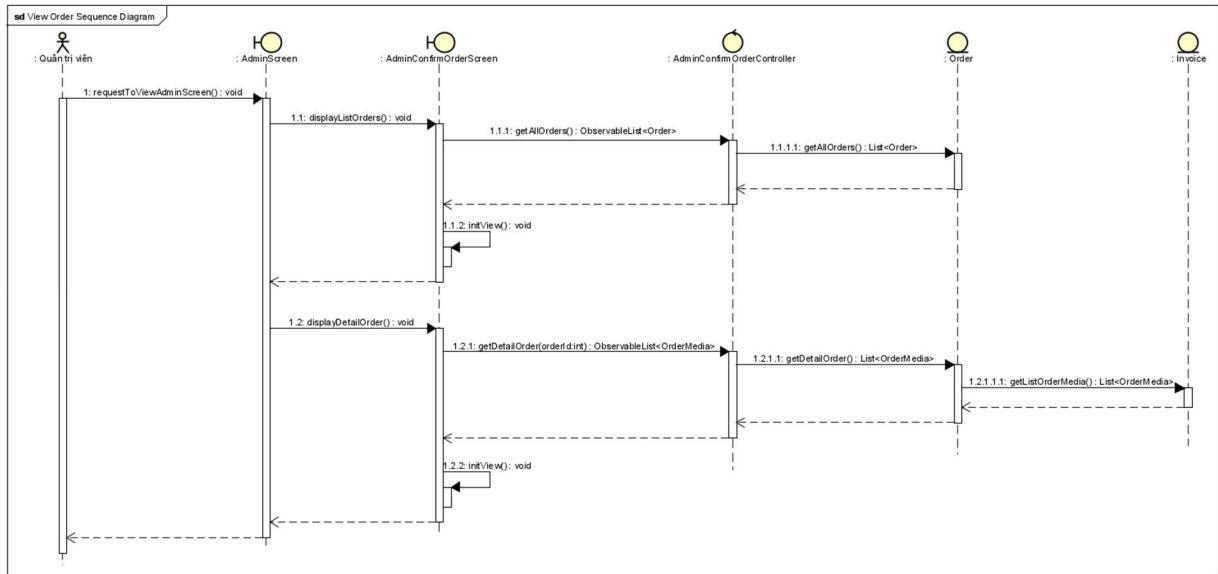


- Hoàn tiền

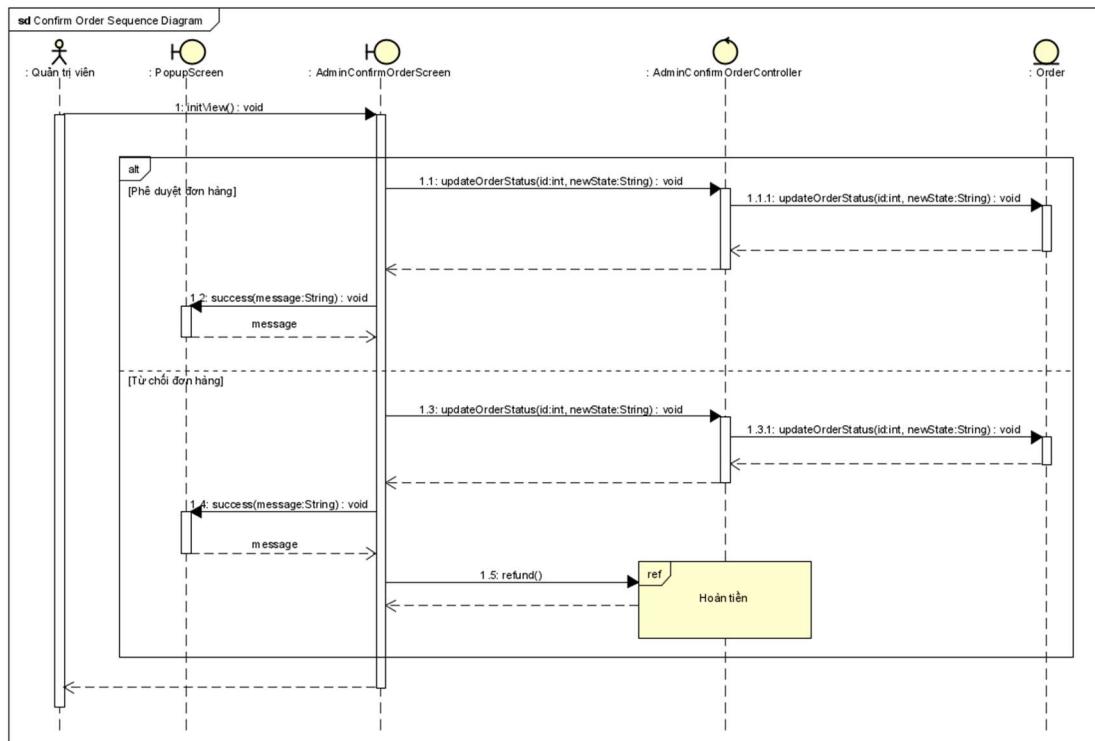


3. Sequence Diagrams

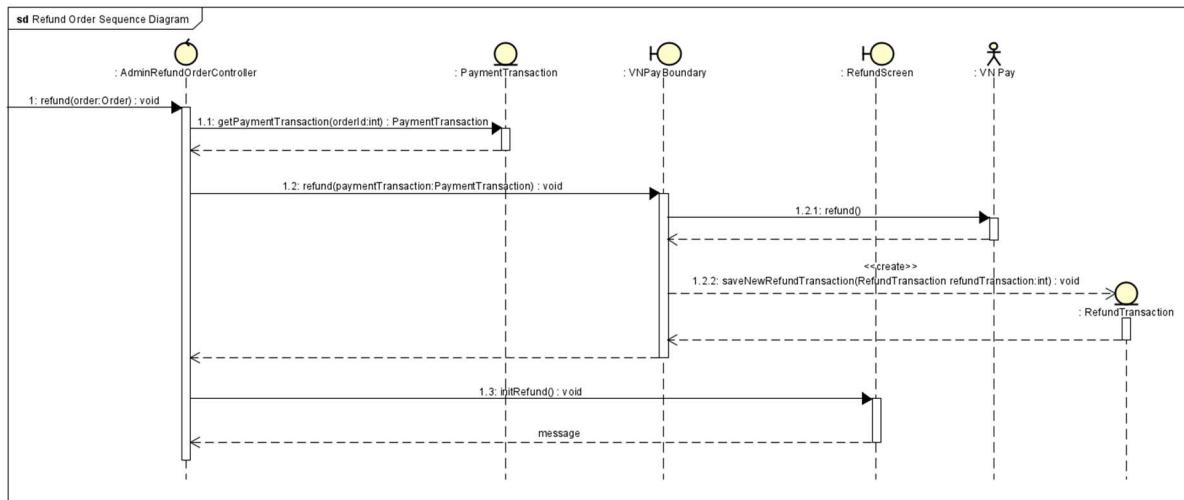
- View Order



- Confirm Order

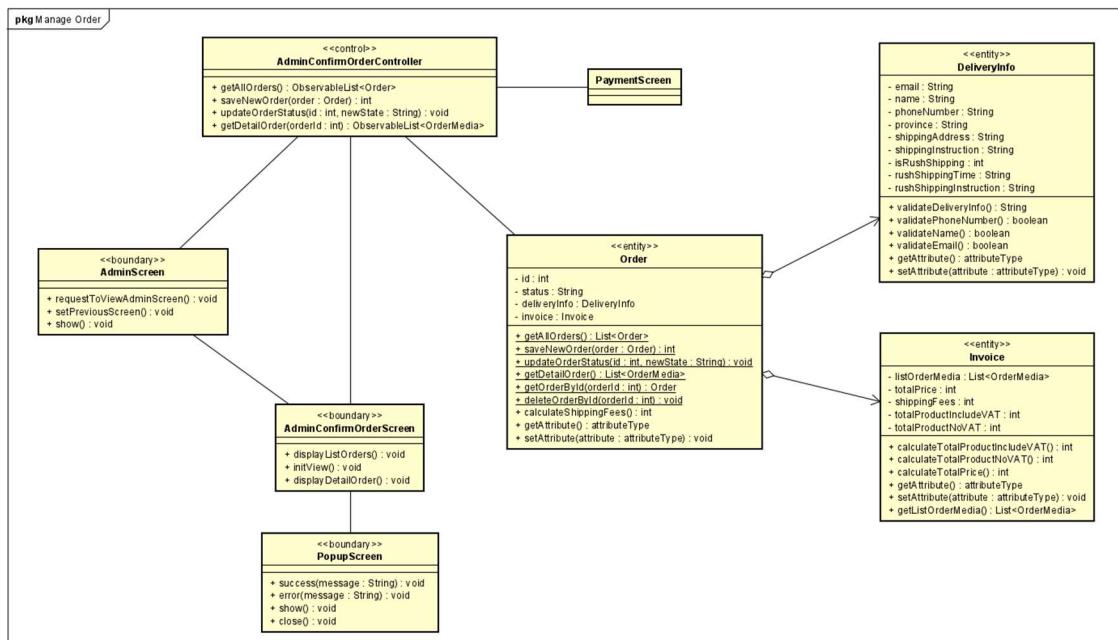


- Refund Order

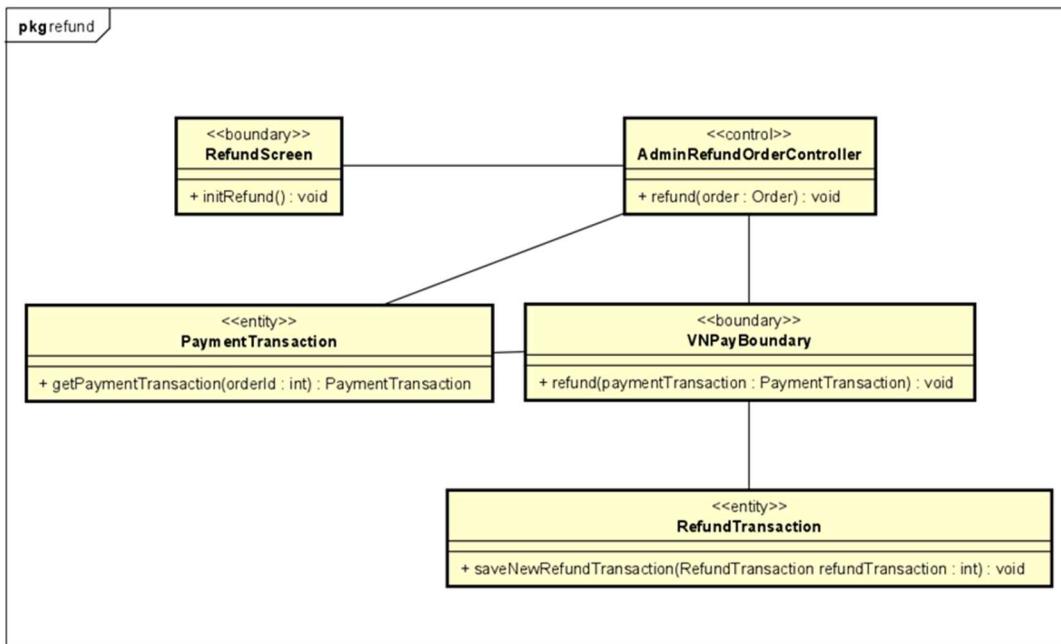


4. Class Diagrams

- View Order and Confirm Order



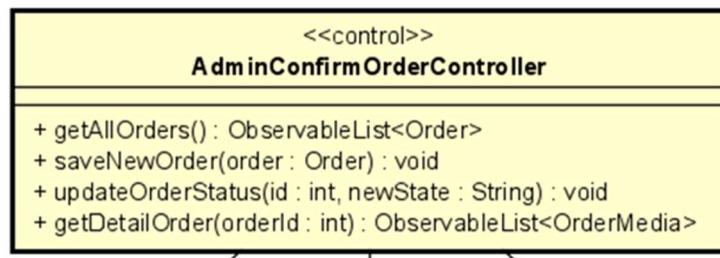
- Refund Order



3.2.1.2. Thiết kế use case Quản lý đơn hàng

Chi tiết Class Diagrams

- Class AdminConfirmOrderController



1. Thuộc tính

Không

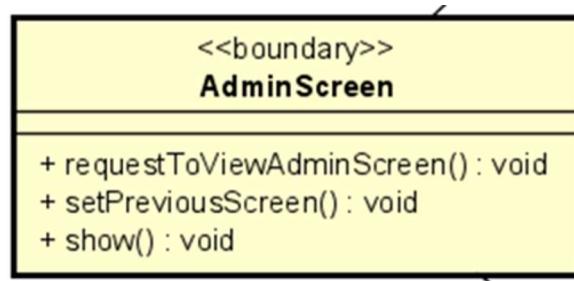
2. Phương thức

#	Name	Return Type	Description
1	getAllOrders	ObservableList<Order>	Lấy ra tất cả các đơn hàng đã đặt còn lưu trong hệ thống
2	saveNewOrder	void	Tạo mới một đơn hàng sau đó lưu vào hệ thống
3	updateOrderStatus	void	Cập nhật trạng thái của đơn hàng được chọn
4	getDetailOrder	ObservableList<OrderMedia>	Lấy ra chi tiết các sản phẩm trong 1 đơn hàng cụ thể

Tham số:

- order – đơn hàng được chọn (được lấy ra)
- id – id của đơn hàng được chọn
- newState – trạng thái mới của đơn hàng
- orderId – id của đơn hàng được chọn

- Class AdminScreen



1. Thuộc tính

Không

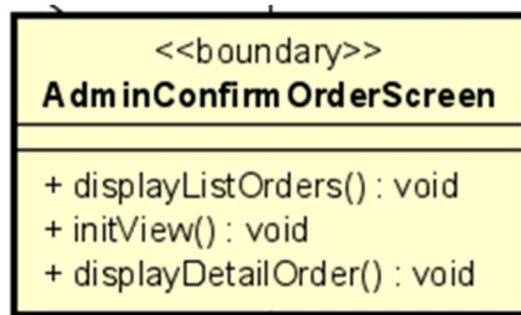
2. Phương thức

#	Name	Return Type	Description
1	requestToViewAdminScreen	void	Yêu cầu mở màn AdminScreen
2	setPreviousScreen	void	Quay trở về màn hình trước đó
3	show	void	Hiển thị màn hình

Tham số

Không

- Class AdminConfirmOrderScreen



1. Thuộc tính

Không

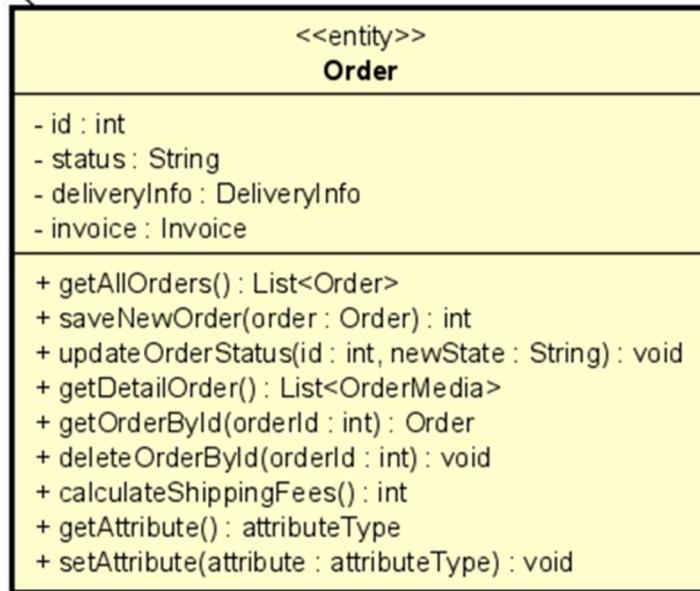
2. Phương thức

#	Name	Return Type	Description
1	displayListOrders	void	Lấy ra và hiển thị các trường yêu cầu vào bảng danh sách đơn hàng
2	initView	void	Khởi tạo màn Admin Order sau đó hiển thị danh sách đơn hàng
3	displayDetailOrder	void	Lấy ra và hiển thị các trường yêu cầu vào bảng chi tiết đơn đặt hàng

Tham số:

Không

- Class Order



1. Thuộc tính

- Id - id đơn hàng
- status - trạng thái đơn hàng
- deliveryInfo - thông tin giao hàng
- invoice - hóa đơn

2. Phương thức

#	Name	Return Type	Description
1	getAllOrders	List<Order>	Lấy ra tất cả các đơn hàng đã đặt còn lưu trong hệ thống
2	saveNewOrder	int	Tạo mới một đơn hàng sau đó lưu vào hệ thống
3	updateOrderStatus	void	Cập nhật trạng thái của đơn hàng được chọn
4	getDetailOrder	List<OrderMedia>	Lấy ra chi tiết các sản phẩm trong 1 đơn hàng cụ thể
5	getOrderById	Order	Lấy ra 1 đơn hàng với id tương ứng
6	deleteOrderById	void	Xóa đơn hàng với id tương ứng
7	calculateShippingFees	int	Tính phí giao hàng

8	getAttribute	attributeTye	Getter
9	setAttribute	void	Setter

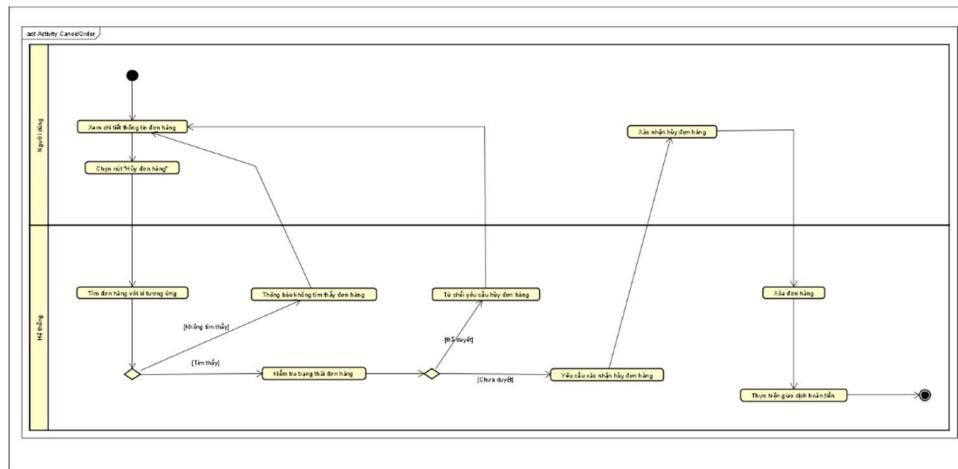
Tham số:

- order – đơn hàng được chọn (được lấy ra)
- id – id của đơn hàng được chọn
- newState – trạng thái mới của đơn hàng
- orderId – id của đơn hàng được chọn

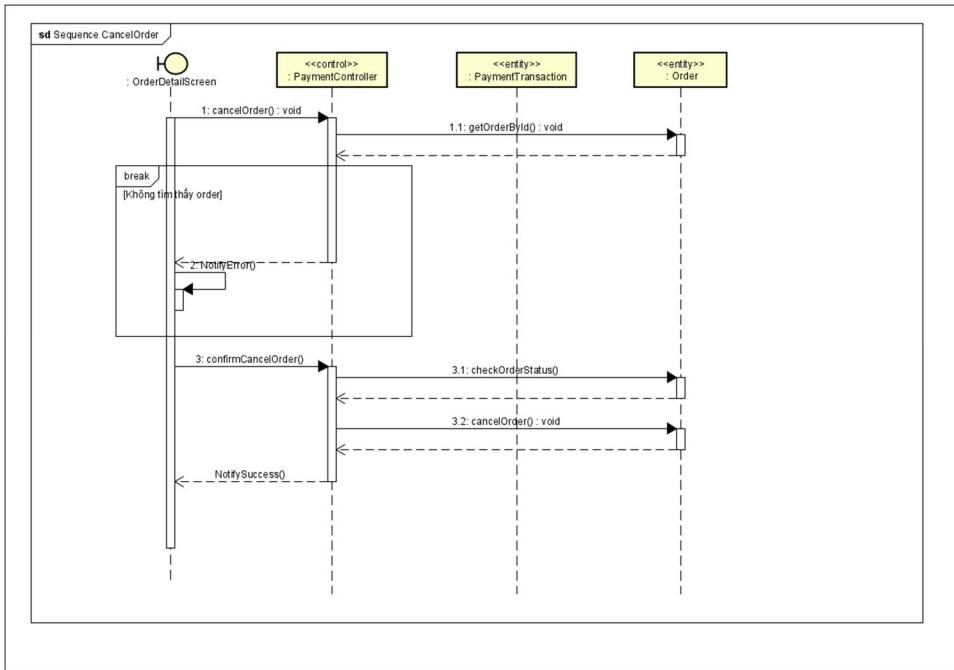
3.2.2. Use case Hủy đơn hàng (Nguyễn Cao Bảo Hiếu)

3.2.2.1. Phân tích use case Hủy đơn hàng

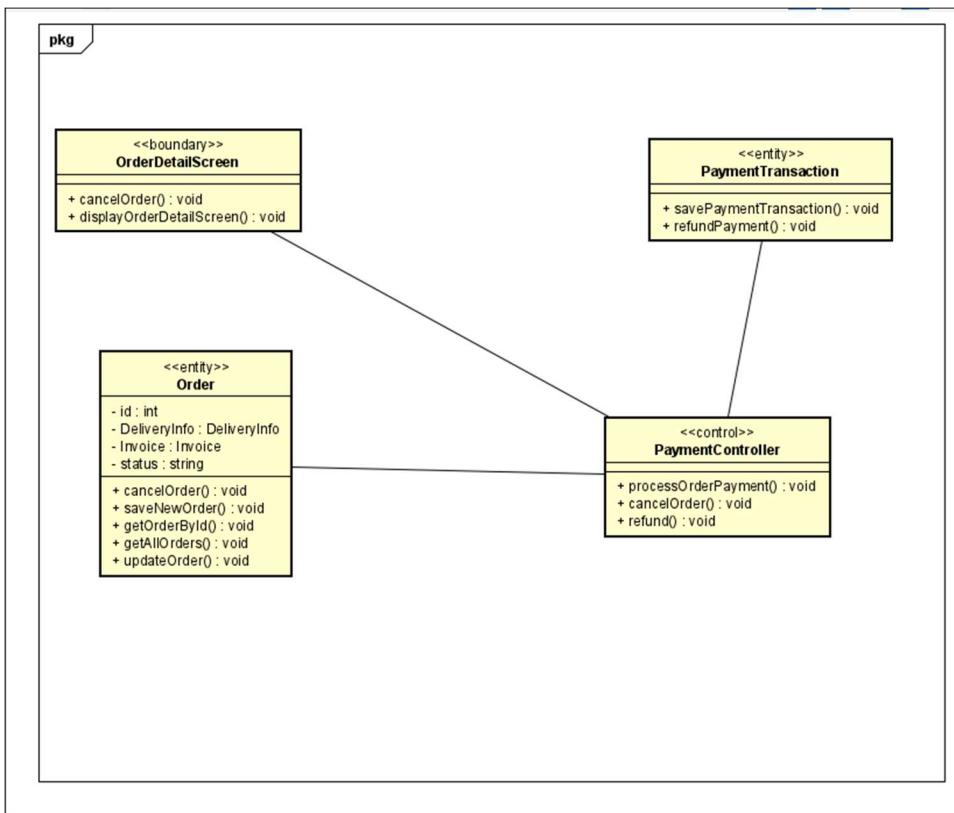
1. Activity Diagram:



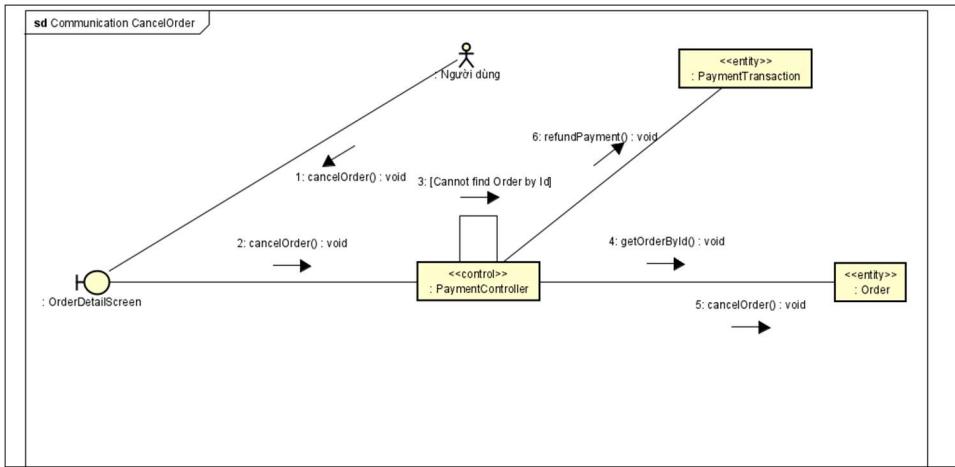
2. Sequence Diagram:



3. Class Diagram:

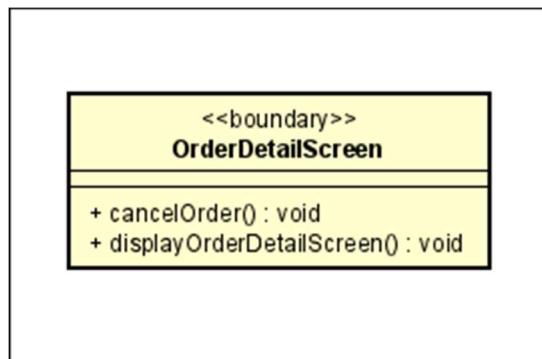


3. Communication Diagram:



3.2.2.2. Thiết kế usecase Hủy đơn hàng:

1. OrderDetailScreen:



1. Thuộc tính

Không

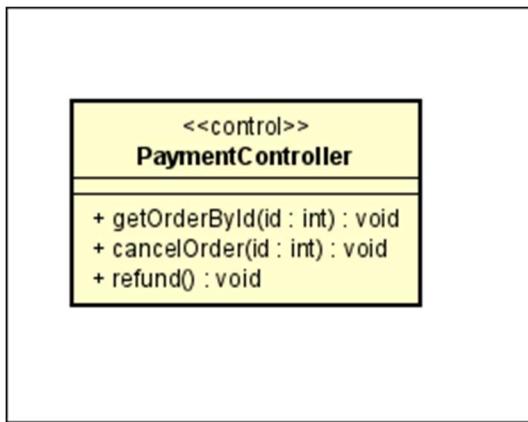
2. Phương thức

#	Name	Return Type	Description
1	displayOrderDetailScreen	void	Lấy ra và hiển thị các trường yêu cầu vào bảng danh sách các thông tin thanh toán của đơn hàng
2	cancelOrder	void	Gửi yêu cầu hủy đơn hàng của khách hàng

Tham số:

Không

3.PaymentController:



1.Thuộc tính

- Id - id đơn hàng

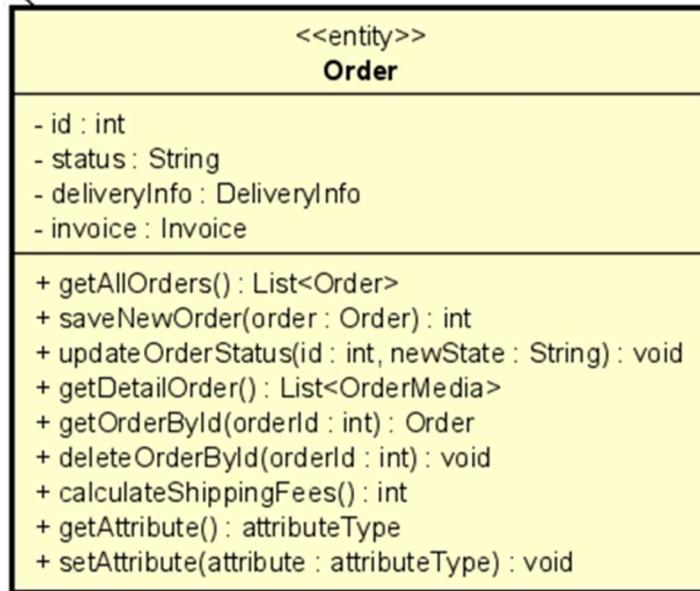
2. Phương thức

#	Name	Return Type	Description
1	getOrderById	void	Lấy ra thông tin cần thiết của đơn hàng theo id được gửi theo email
2	cancelOrder	void	Hủy đơn hàng theo id đã được gửi qua email
3	refund	void	Hoàn tiền theo đơn thanh toán

3.Tham số:

Không

3.Class Order:



4. Thuộc tính

- id - id đơn hàng
- status - trạng thái đơn hàng
- deliveryInfo - thông tin giao hàng
- invoice - hóa đơn

5. Phương thức

#	Name	Return Type	Description
1	getAllOrders	List<Order>	Lấy ra tất cả các đơn hàng đã đặt còn lưu trong hệ thống
2	saveNewOrder	int	Tạo mới một đơn hàng sau đó lưu vào hệ thống
3	updateOrderStatus	void	Cập nhật trạng thái của đơn hàng được chọn
4	getDetailOrder	List<OrderMedia>	Lấy ra chi tiết các sản phẩm trong 1 đơn hàng cụ thể
5	getOrderById	Order	Lấy ra 1 đơn hàng với id tương ứng
6	deleteOrderById	void	Xóa đơn hàng với id tương ứng
7	calculateShippingFees	int	Tính phí giao hàng

8	getAttribute	attributeTye	Getter
9	setAttribute	void	Setter

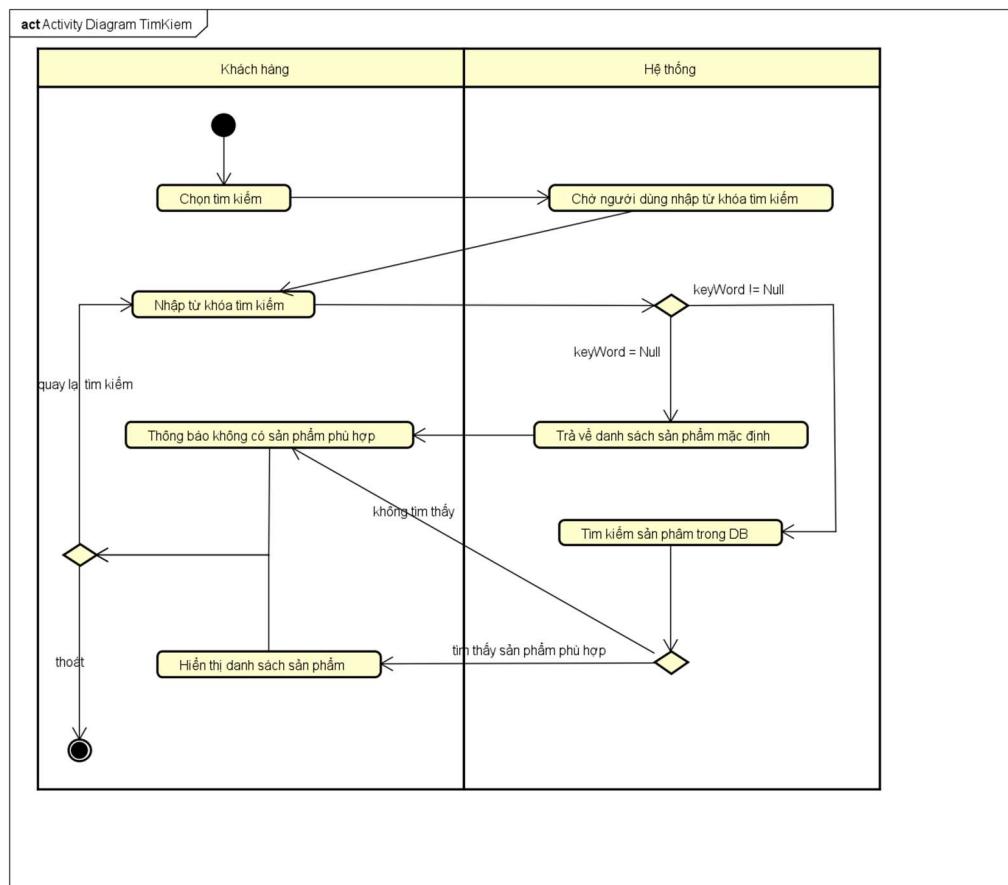
Tham số:

- order – đơn hàng được chọn (được lấy ra)
 - id – id của đơn hàng được chọn
 - newState – trạng thái mới của đơn hàng
 - orderId – id của đơn hàng được chọn

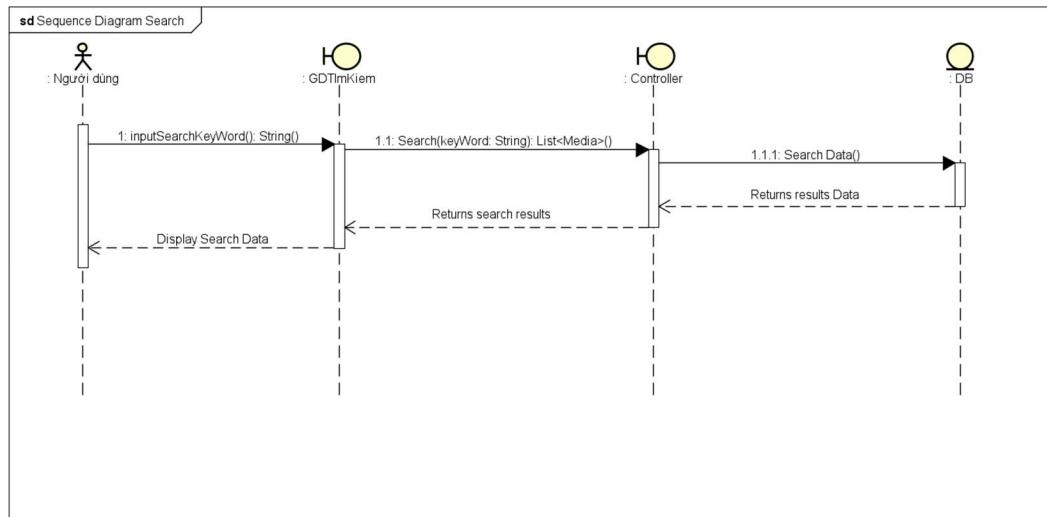
3.2.3. Use case Sắp xếp sản phẩm và Tìm kiếm sản phẩm (Nguyễn Đình Hiếu)

3.2.3.1. Phân tích usecase tìm kiếm sản phẩm

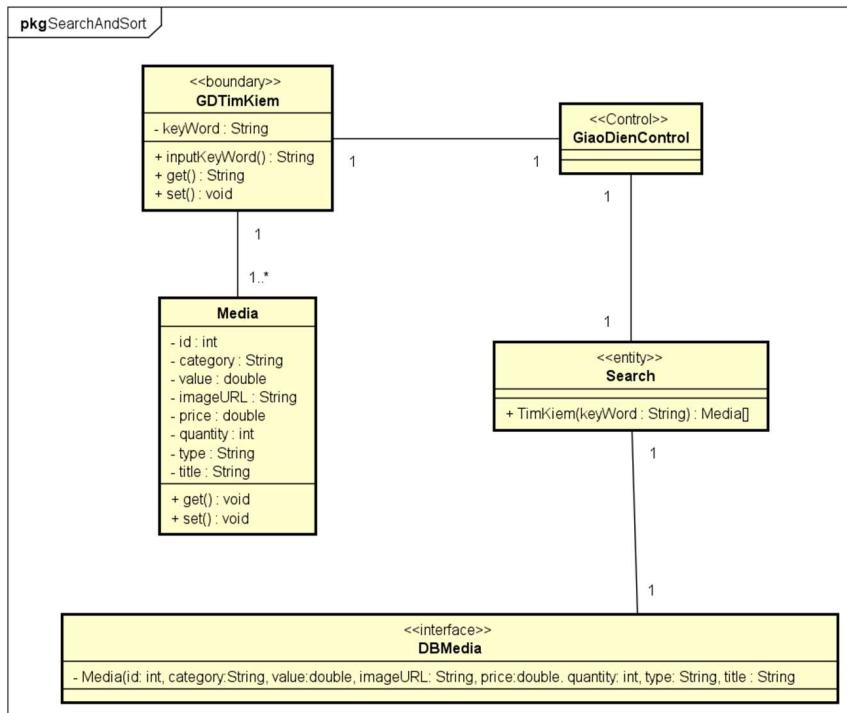
3.2.3.1.1. Activity Diagrams



3.2.3.1.2. Sequence Diagrams

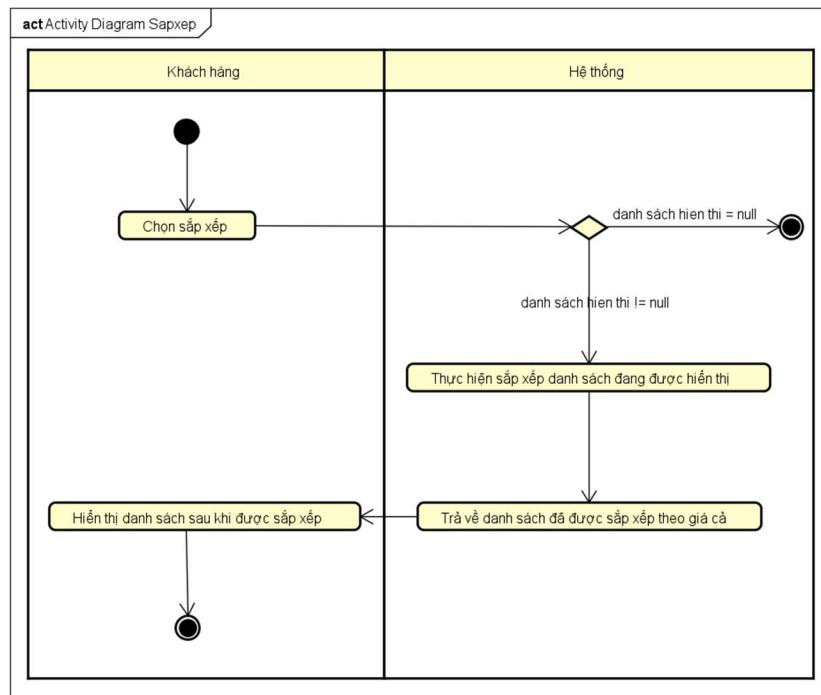


3.2.3.1.3. Analysis Class Diagrams

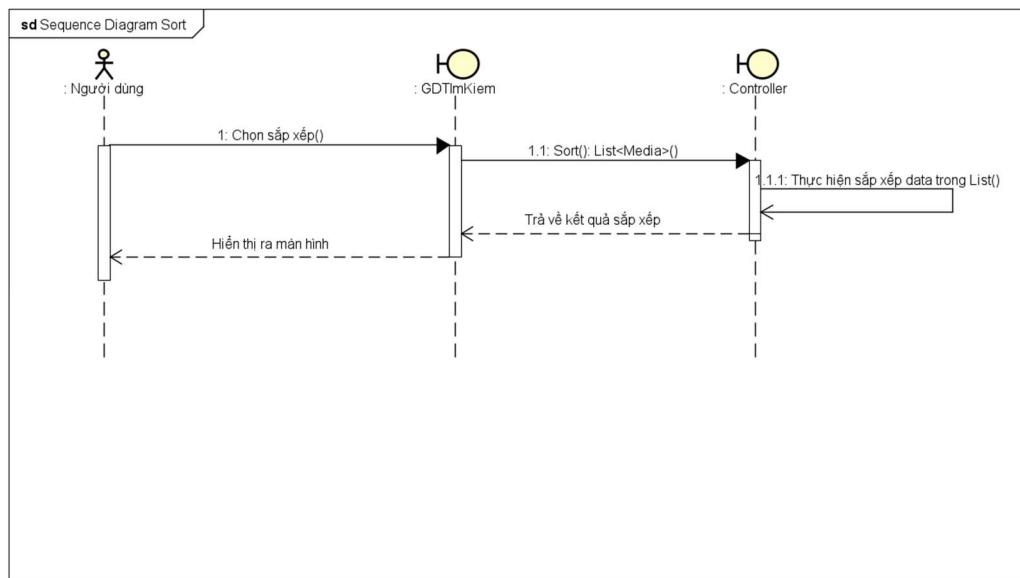


3.2.3.2. Phân tích usecase sáp xếp sản phẩm

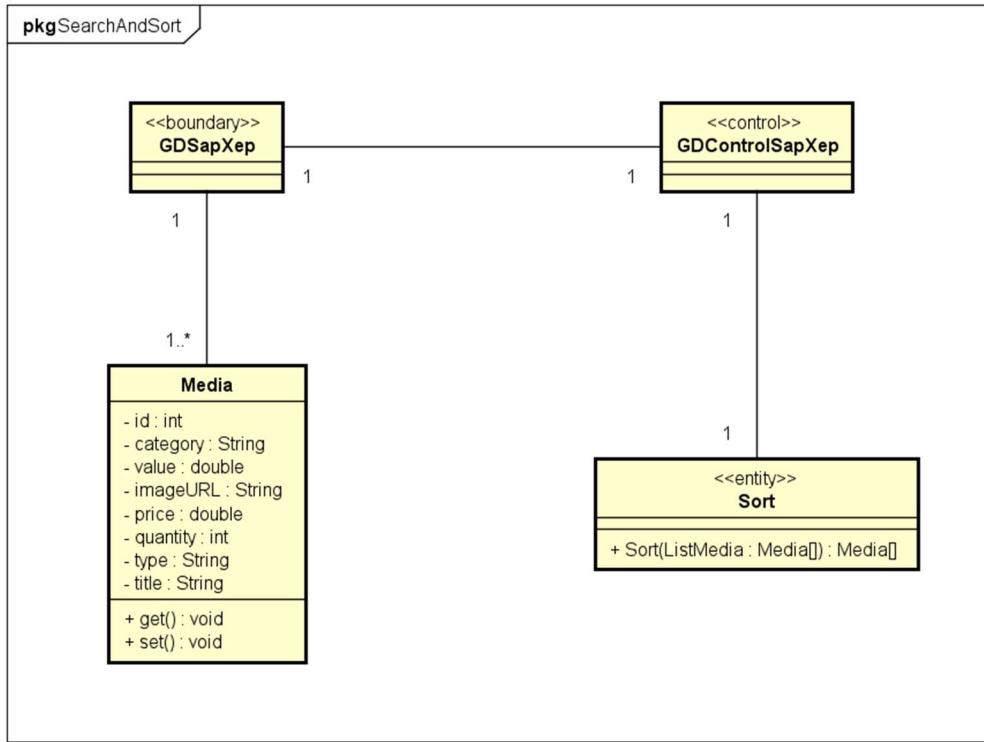
3.2.3.2.1. Activity Diagrams



3.2.3.2.2. Sequence Diagrams

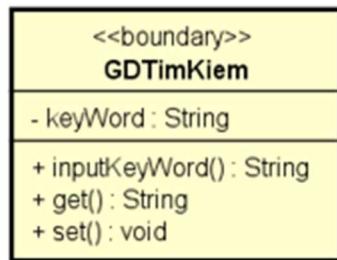


3.2.3.2.3. Analysis Class Diagrams



3.2.3.3. Thiết kế usecase tìm kiếm sản phẩm

* Class **GDTimKiem**



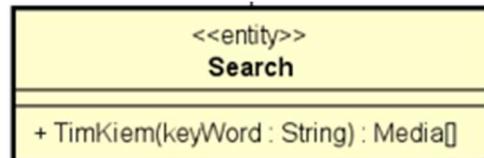
1. Thuộc tính

- **keyWord**: thông tin sản phẩm người dùng muốn tìm kiếm

2. Phương thức

- **inputKeyWord()**: người dùng nhập đầu vào cho **keyWord**

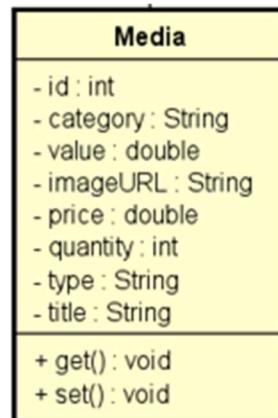
* Class Search



Phương thức:

- Tìm kiếm: Thực hiện tìm kiếm các sản phẩm tương đồng với keyWord được người dùng nhập vào và trả về 1 danh sách các sản phẩm phù hợp.

*Class media:

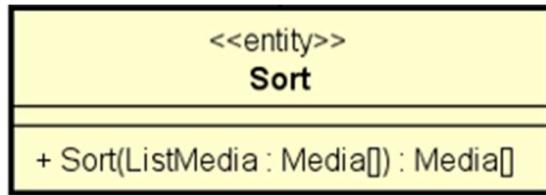


Thuộc tính:

- Id : Id của sản phẩm
- Category: thể loại của sản phẩm (nội dung thuộc thể loại gì)
- imageURL: ảnh của sản phẩm
- Price: giá của sản phẩm
- Quantity: số lượng sản phẩm
- Type: kiểu bản cứng là gì (book, cd, dvd)
- Title: tên sản phẩm

3.2.3.4. Thiết kế usecase sắp xếp sản phẩm

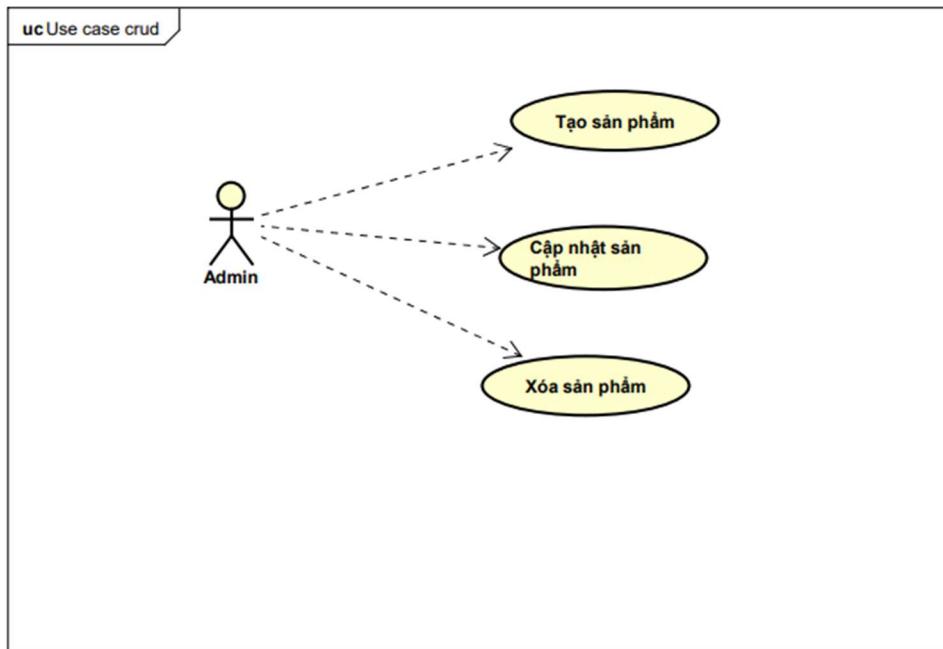
* Class Sort



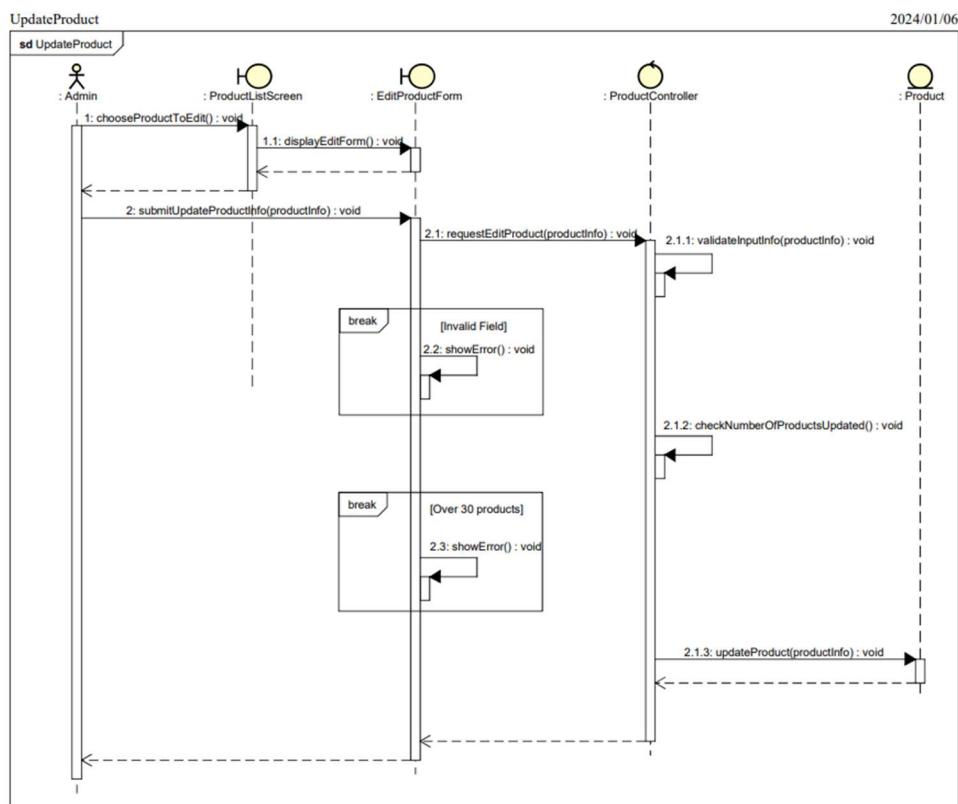
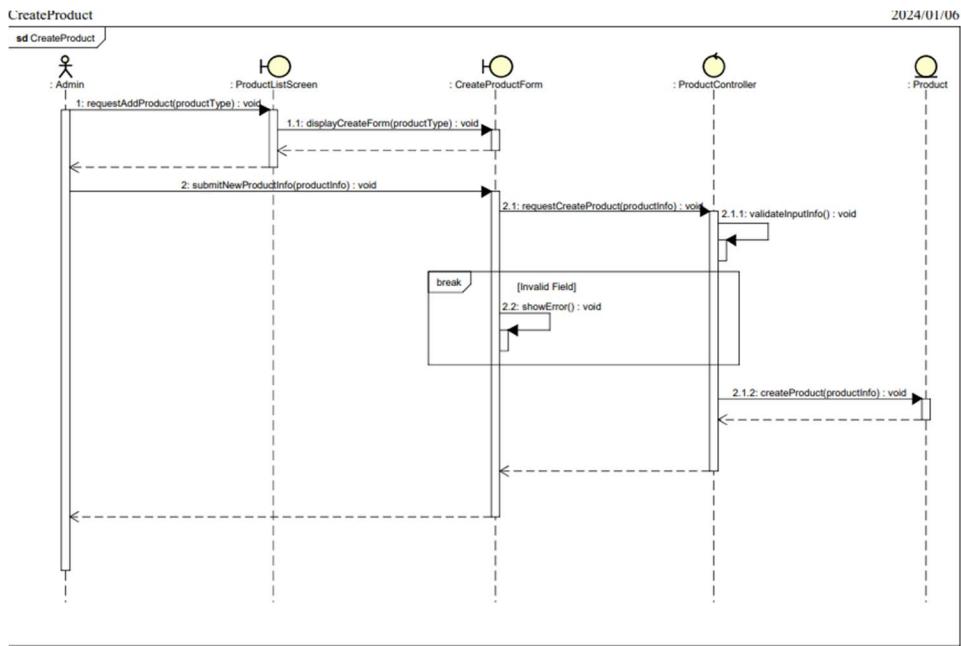
Phương thức Sort: Tham số ListMedia với kiểu là 1 danh sách các Media được hiển thị trên màn hình, phương thức sẽ thực hiện sắp xếp các Media theo giá cả từ bé tới lớn và sau đó sẽ hiển thị lại các Media lên màn hình. Phương thức trả về 1 danh sách các Media sau khi sắp xếp.

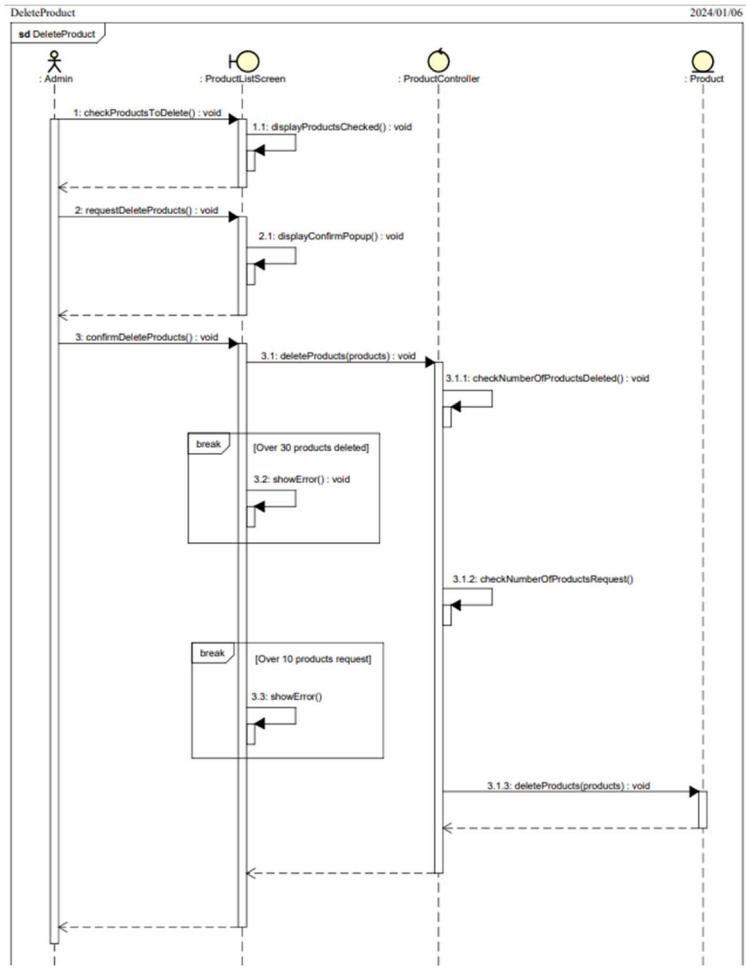
3.2.4. Use case CRUD sản phẩm (Phạm Minh Hiếu)

3.2.4.1 Use case



3.2.4.2 Sequence Diagram (Thêm, sửa, xóa sản phẩm)





3.2.4.3 Class diagram

