

HANOI UNIVERSITY OF SCIENCE AND TECHNOLOGY

School of Information and communications technology

Software Design Document

AN INTERNET MEDIA STORE

Subject: Thiết kế xây dựng phần mềm

Nhóm 17

Nguyễn Cảnh Phước - 20215456

Nguyễn Minh Phương – 20215458

Vũ Đức Anh – 20193985

Vũ Việt Phương – 20215459

Bùi Trung Quân – 20194142

Giang Trung Quân - 20215463

Hanoi, <01, 2025>

<All notations inside the angle bracket are not part of this document, for its purpose is for extra instruction. When using this document, please erase all these notations and/or replace them with corresponding content as instructed>

<This document, written by Prof. NGUYEN Thi Thu Trang, is used as a case study for student with related courses. Any modifications and/or utilization without the consent of the author is strictly forbidden>

4.4.1.2 Class Diagram for Package Controller

Table of Contents

Table of Contents	3
1 Introduction	7
1.1 Objective.....	7
1.2 Scope	7
1.3 Glossary	7
1.4 References.....	10
2 Overall Description.....	11
2.1 General Overview	11
2.2 Assumptions/Constraints/Risks	12
2.2.1 Assumptions	12
2.2.2 Constraints.....	12
2.2.3 Risks.....	12
3 System Architecture and Architecture Design	13
3.1 Architectural Patterns	13
3.2 Interaction Diagrams	13
3.3 Analysis Class Diagrams	21
3.4 Unified Analysis Class Diagram	24
3.5 Security Software Architecture.....	24
4 Detailed Design	25
4.1 User Interface Design.....	25
4.1.1 Screen Configuration Standardization	25
4.1.2 Screen Transition Diagrams	25
4.1.3 Screen Specifications	26

4.2	Data Modeling.....	32
4.2.1	Conceptual Data Modeling.....	32
4.2.2	Database Design	32
4.3	Non-Database Management System Files	46
4.4	Class Design.....	47
4.4.1	General Class Diagram.....	47
4.4.2	Class Diagrams	48
4.4.3	Class Design	Error! Bookmark not defined.
5	Design Considerations	1
5.1	Goals and Guidelines.....	1
5.2	Architectural Strategies	1
5.3	Coupling and Cohesion	2
5.4	Design Principles.....	2
5.5	Design Patterns.....	2

List of Figures

No table of figures entries found.

List of Tables

No table of figures entries found.

1 Introduction

Áp dụng kiến thức từ môn học, nhóm chúng em phát triển ứng dụng từ bước yêu cầu, bước phân tích, thiết kế đến bước triển khai và giai đoạn thử nghiệm. Tài liệu này cung cấp một cái nhìn tổng quan về kiến trúc của hệ thống, sử dụng nhiều góc nhìn kiến trúc để minh họa các thành phần hệ thống khác nhau. Nó nhằm mục đích áp dụng các phần mềm quan trọng kiến thức kiến trúc đã được nghiên cứu.

1.1 Objective

Mục tiêu của tài liệu thiết kế phần mềm (SDD) này là mô tả chi tiết thiết kế của hệ thống AIMS (cụ thể là chức năng duyệt sản phẩm, và các chức năng liên quan đến quản lý giỏ hàng), bao gồm kiến trúc hệ thống, thiết kế giao diện người dùng, thiết kế cơ sở dữ liệu, và thiết kế chi tiết của các lớp. Tài liệu này dành cho các stakeholders của dự án bao gồm: developers, testers, project managers, và khách hàng.

1.2 Scope

Tài liệu này mô tả thiết kế cho hệ thống AIMS, một hệ thống quản lý bán hàng media (sách, CD, DVD, LP). Cụ thể, tài liệu này tập trung vào các chức năng sau:

- *Duyệt sản phẩm (Browse Products)*
- *Xem chi tiết sản phẩm (View Product Detail)*
- *Đặt hàng*
- *Thanh toán*
- *Quản lý giỏ hàng (Manage Cart) bao gồm:*
 - *Thêm sản phẩm vào giỏ hàng (Add to Cart)*
 - *Xóa sản phẩm khỏi giỏ hàng (Remove from Cart)*
 - *Cập nhật số lượng sản phẩm (Update Quantity)*
 - *Chọn sản phẩm để đặt hàng.*
 - *Tiến hành đặt hàng*

1.3 Glossary

Thuật ngữ	Định nghĩa
AIMS	Tên của hệ thống đang được thiết kế.
Media	Các sản phẩm được bán trong hệ thống, bao gồm Book, CD, DVD, LP.
Cart	Giỏ hàng, nơi lưu trữ các sản phẩm mà người dùng đã chọn mua.
CartMedia	Lớp đại diện cho một sản phẩm trong giỏ hàng.
Order	Đơn hàng, chứa thông tin về các sản phẩm được đặt mua, thông tin giao hàng, và thông tin thanh toán.
OrderMedia	Lớp đại diện cho một sản phẩm trong đơn hàng.
Invoice	Hóa đơn, chứa thông tin chi tiết về đơn hàng.
DeliveryInfo	Thông tin giao hàng.
RushDeliveryInfo	Thông tin giao hàng nhanh.
PaymentTransaction	Giao dịch thanh toán.
RefundTransaction	Giao dịch hoàn tiền.
User	Người dùng.
Admin	Quản trị viên.
ProductManager	Người quản lý sản phẩm.
SRP	Single Responsibility Principle (Nguyên tắc đơn trách nhiệm).

OCP	Open/Closed Principle (Nguyên tắc mở/đóng).
LSP	Liskov Substitution Principle (Nguyên tắc thay thế Liskov).
ISP	Interface Segregation Principle (Nguyên tắc phân tách interface).
DIP	Dependency Inversion Principle (Nguyên tắc đảo ngược phụ thuộc).
Coupling	Mức độ phụ thuộc giữa các module.
Cohesion	Mức độ liên quan giữa các thành phần trong một module.
SOLID	Nhóm các nguyên tắc thiết kế hướng đối tượng: SRP, OCP, LSP, ISP, DIP.
Factory Pattern	Design pattern thuộc nhóm Creational, cung cấp một interface để tạo các đối tượng mà không cần chỉ định lớp cụ thể của chúng.
Dependency Injection	Kỹ thuật đưa các dependency vào một object thay vì để object tự tạo các dependency đó.
API	Application Programming Interface (Giao diện lập trình ứng dụng).
SQL	Structured Query Language (Ngôn ngữ truy vấn có cấu trúc).
DBMS	Database Management System (Hệ quản trị cơ sở dữ liệu).
UI	User Interface (Giao diện người dùng).
UX	User Experience (Trải nghiệm người dùng).

JSON	JavaScript Object Notation (Định dạng dữ liệu dạng văn bản, dựa trên một tập con của ngôn ngữ lập trình JavaScript).
XML	Extensible Markup Language (Ngôn ngữ đánh dấu mở rộng).
UML	Unified Modeling Language (Ngôn ngữ mô hình hóa thống nhất).
Use Case	Biểu đồ ca sử dụng
Class Diagram	Biểu đồ lớp
Activity Diagram	Biểu đồ hoạt động
Sequence Diagram	Biểu đồ tuần tự
SDD	Software Design Document (Tài liệu thiết kế phần mềm)
SRS	Software Requirements Specification (Tài liệu đặc tả yêu cầu phần mềm)

1.4 References

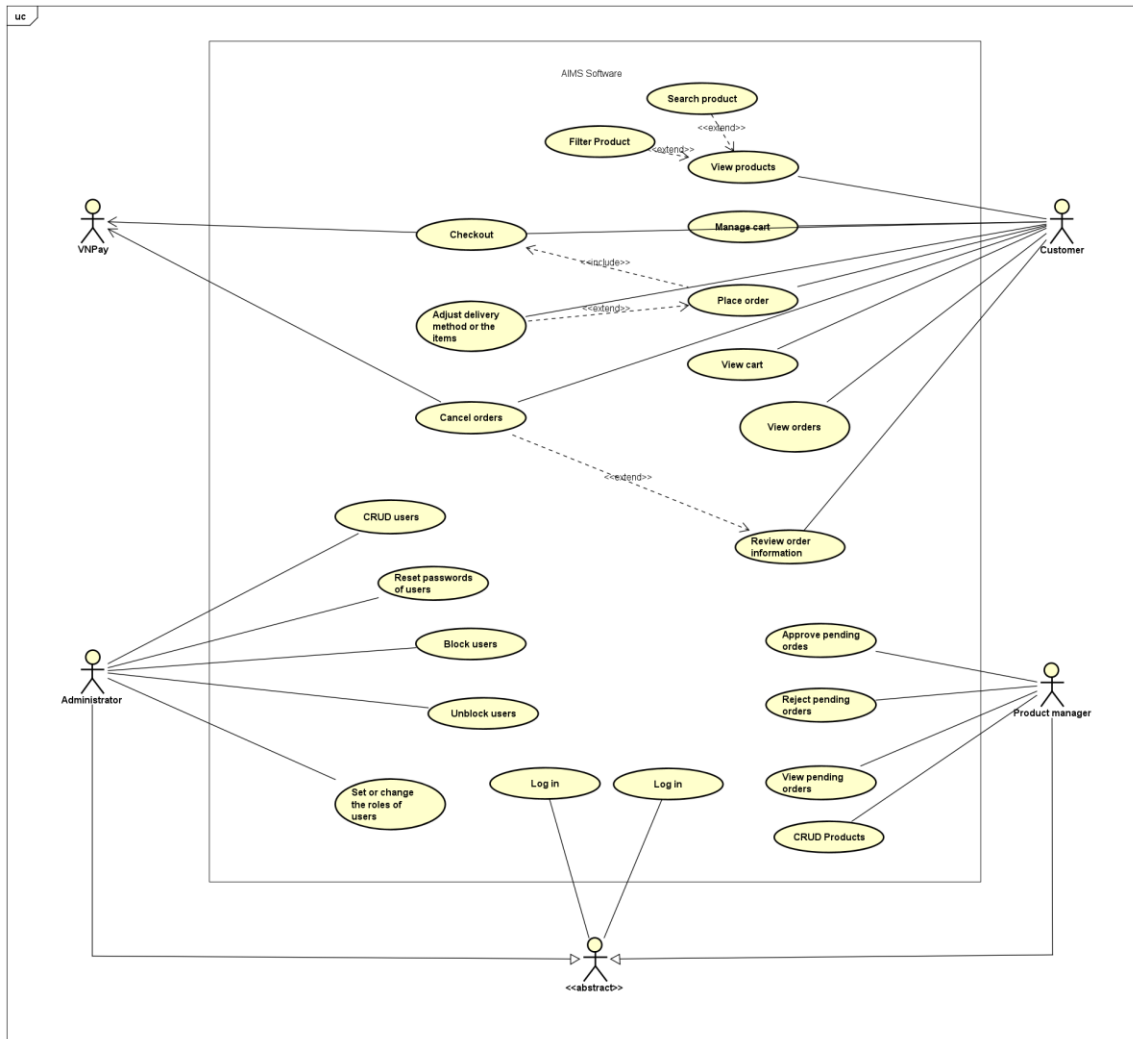
- [1] Centers for Medicare & Medicaid Services, "System Design Document Template," [Online]. Available: <https://www.cms.gov/Research-Statistics-Data-and-Systems/CMS-Information-Technology/XLC/Downloads/SystemDesignDocument.docx>.

<Listing the referenced material used in this document, including the one related to the project>

2 Overall Description

Phần này mô tả các nguyên tắc và chiến lược được sử dụng làm hướng dẫn khi thiết kế và triển khai hệ thống.

2.1 General Overview



Hệ thống AIMS là một hệ thống thương mại điện tử cho phép người dùng mua các sản phẩm media (sách, CD, DVD, LP). Hệ thống được thiết kế theo kiến trúc hướng đối tượng, sử dụng ngôn ngữ lập trình Java và cơ sở dữ liệu MySQL.

2.2 Assumptions/Constraints/Risks

2.2.1 Assumptions

Người dùng phần mềm phải có kết nối Internet ổn định. Ngoài ra, vì chương trình của chúng tôi là một ứng dụng máy tính để bàn, người dùng cũng cần có máy tính xách tay hoặc máy tính để bàn có hệ điều hành (chúng tôi khuyên dùng Microsoft Windows 10 64 bit trở lên, macOS 10.13 trở lên hoặc bất kỳ bản phân phối Linux nào cho phép chạy ứng dụng) để sử dụng các ứng dụng. Về yêu cầu hệ thống, chúng tôi muốn nói rằng bạn cần ít nhất 2 GB RAM và ổ cứng thể rắn có ít nhất 3 GB dung lượng trống. Phiên bản JRE mới nhất cũng được yêu cầu và độ phân giải màn hình 1280 x 800 là phù hợp.

2.2.2 Constraints

Phần cứng hoặc môi trường phần mềm Ràng buộc: Ứng dụng phải tương thích với các hệ điều hành Windows, macOS và Linux. Tác động: Điều này đòi hỏi các công cụ và thư viện phát triển đa nền tảng, điều này có thể hạn chế việc lựa chọn ngôn ngữ lập trình và framework.

Ràng buộc: Ứng dụng phải thân thiện với người dùng và chỉ yêu cầu các kỹ năng máy tính cơ bản. Tác động: Giao diện người dùng phải trực quan, với các hướng dẫn rõ ràng và điều hướng đơn giản. Điều này sẽ liên quan đến việc kiểm tra khả năng sử dụng rộng rãi và có thể là một thiết kế đơn giản hóa để đảm bảo khả năng tiếp cận cho tất cả người dùng.

2.2.3 Risk

Phát triển phần mềm thương mại điện tử trên máy tính để bàn liên quan đến một số rủi ro, bao gồm các vấn đề tương thích giữa các hệ điều hành, đảm bảo giao diện thân thiện với người dùng cho những người có kỹ năng máy tính cơ bản và quản lý các nguồn lực phát triển hạn chế. Tuân thủ các tiêu chuẩn ngành và tích hợp với các dịch vụ của bên thứ ba là điều cần thiết để tránh các vấn đề pháp lý và hoạt động.

3 System Architecture and Architecture Design

3.1 Architectural Patterns

a. Model-View-Controller (MVC)

Tách biệt logic nghiệp vụ (Model), giao diện người dùng (View), và xử lý tương tác người dùng (Controller).

- Model: Các lớp Entity (Media, Order, Cart, User,...), các lớp Repository.
- View: Các lớp *ScreenController và các file FXML.
- Controller: Các lớp *Controller (CartController, PlaceOrderController,...).

b. Factory Pattern

Sử dụng MediaRepositoryFactory để tạo các đối tượng MediaRepository và MediaScreenControllerFactory để tạo các đối tượng *ScreenController tương ứng với từng loại media.

c. Dependency Injection

Inject các dependency (ví dụ: MediaRepository, CartController) vào các lớp thông qua constructor, giúp giảm coupling và tăng tính linh hoạt.

d. Repository Pattern

Sử dụng các lớp *Repository để truy cập và thao tác với cơ sở dữ liệu.

e. Service Layer Pattern

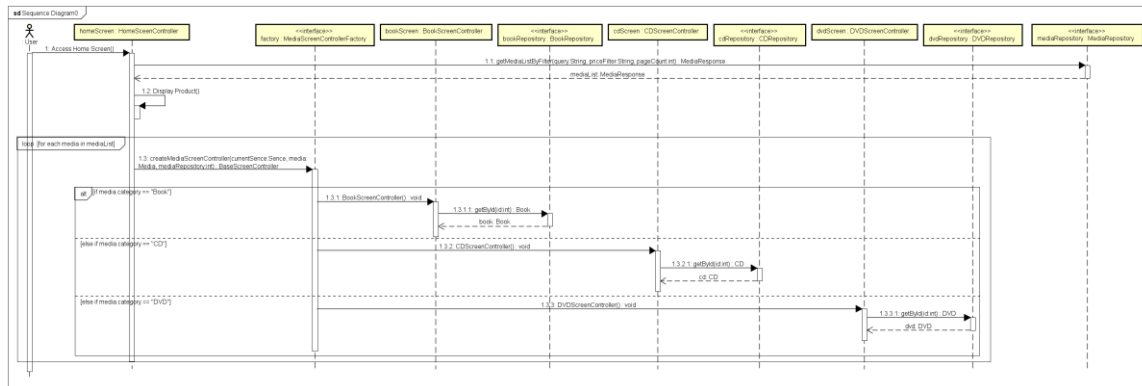
Sử dụng các interface *Service để định nghĩa các dịch vụ của hệ thống (MediaService, OrderService, PaymentService, DeliveryService,...).

f. Singleton Pattern

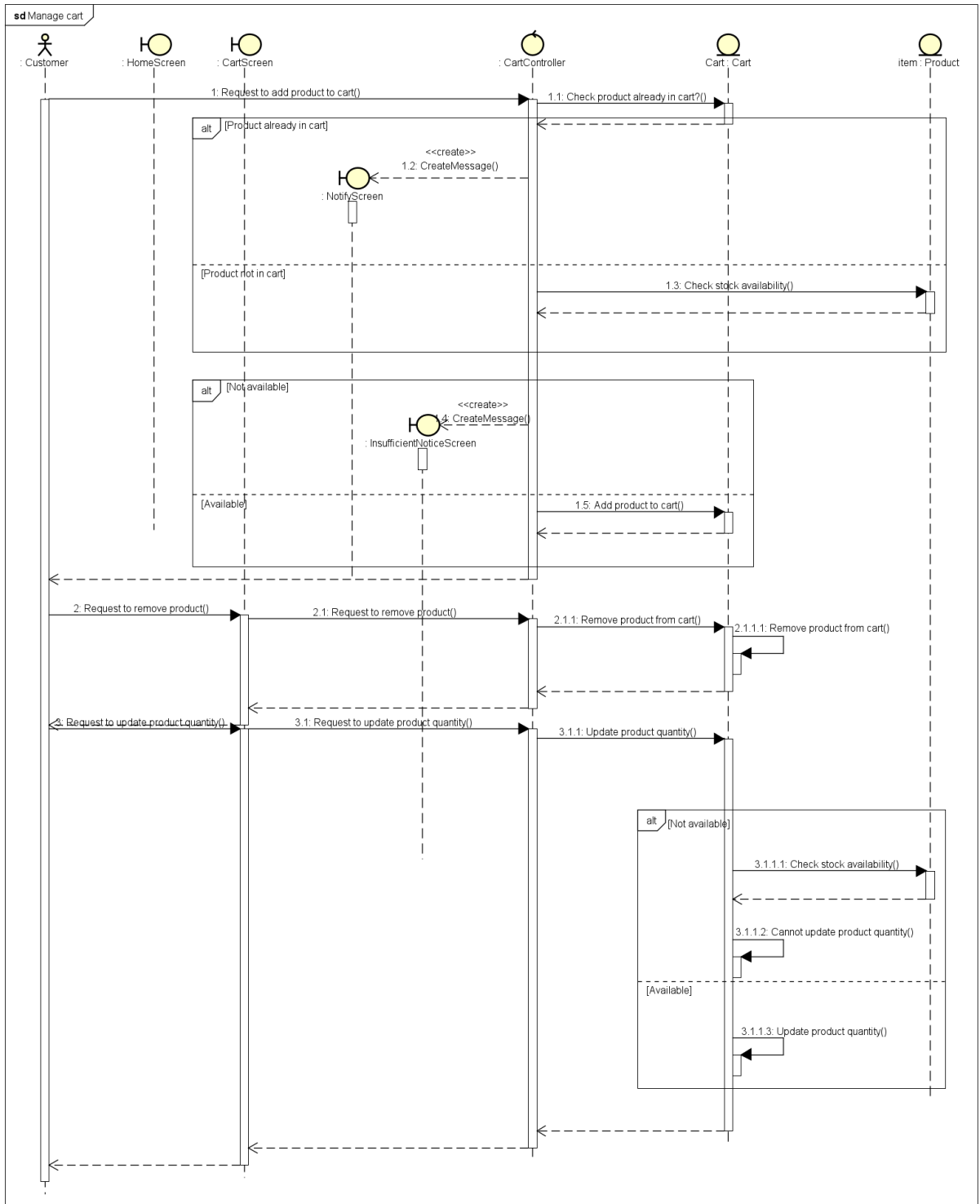
Áp dụng cho lớp Cart để đảm bảo chỉ có một instance của giỏ hàng trong toàn hệ thống của hệ thống

3.2 Interaction Diagrams

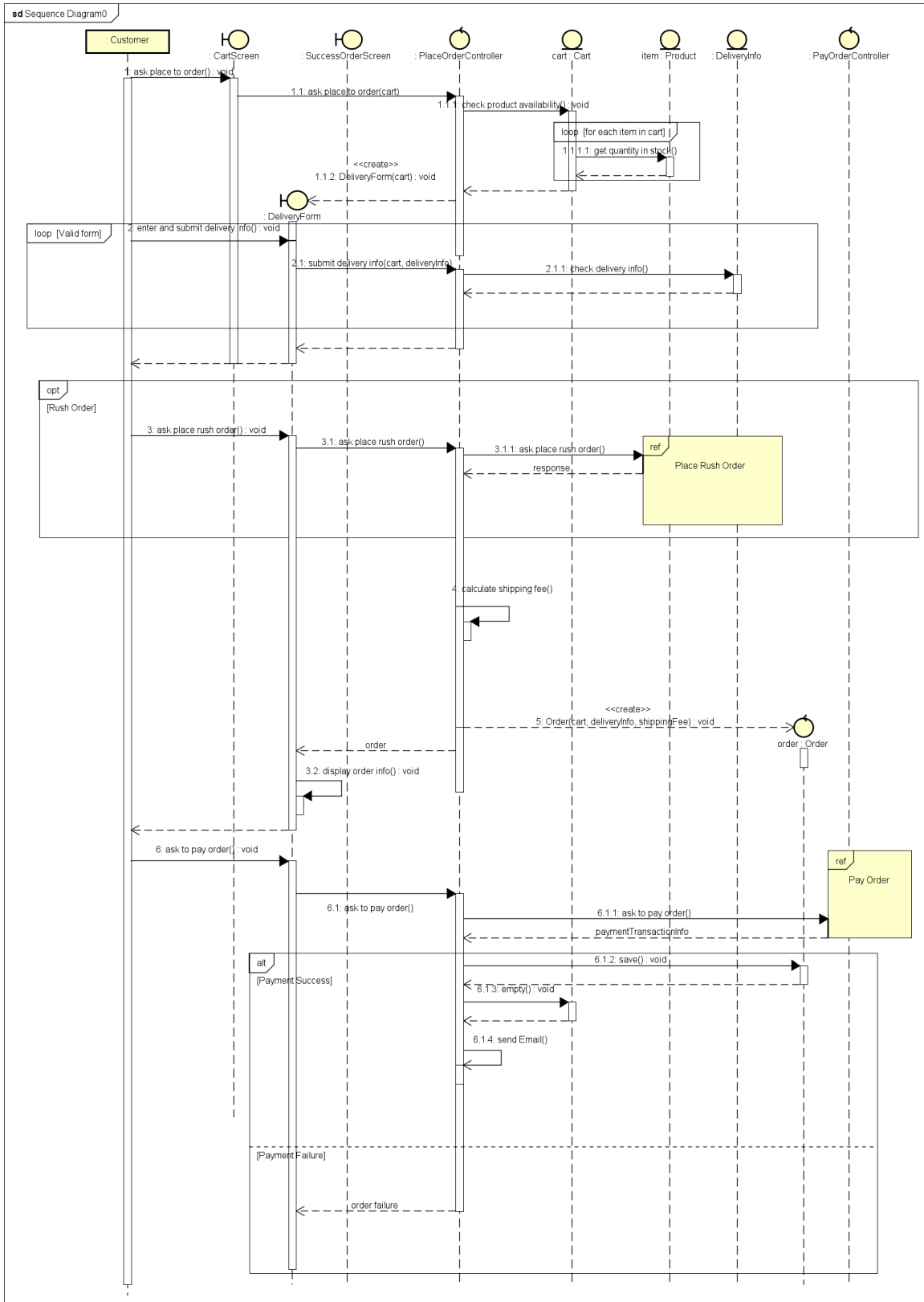
3.2.1 Browse product sequence diagram



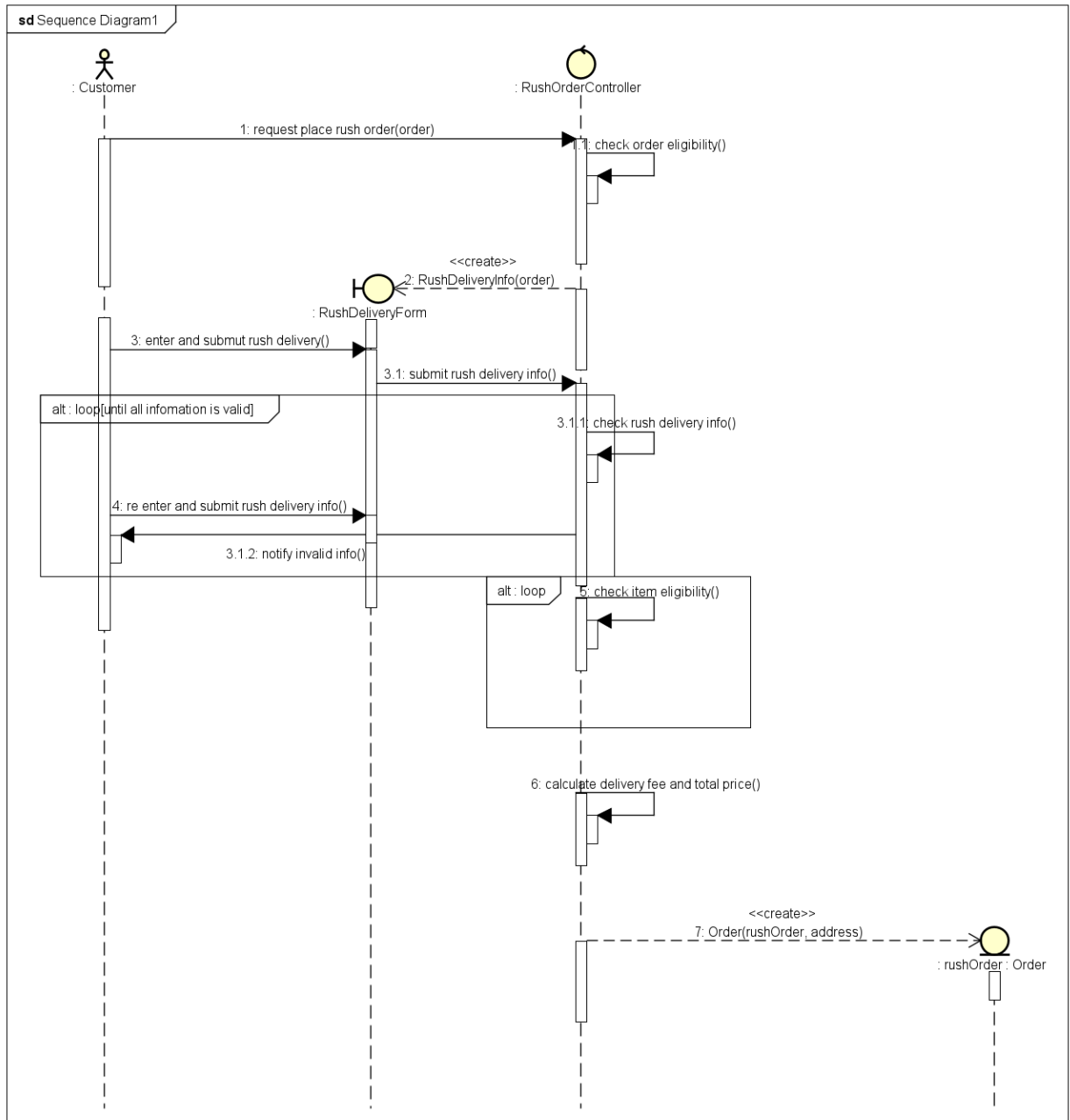
3.2.2 Manage cart sequence diagram



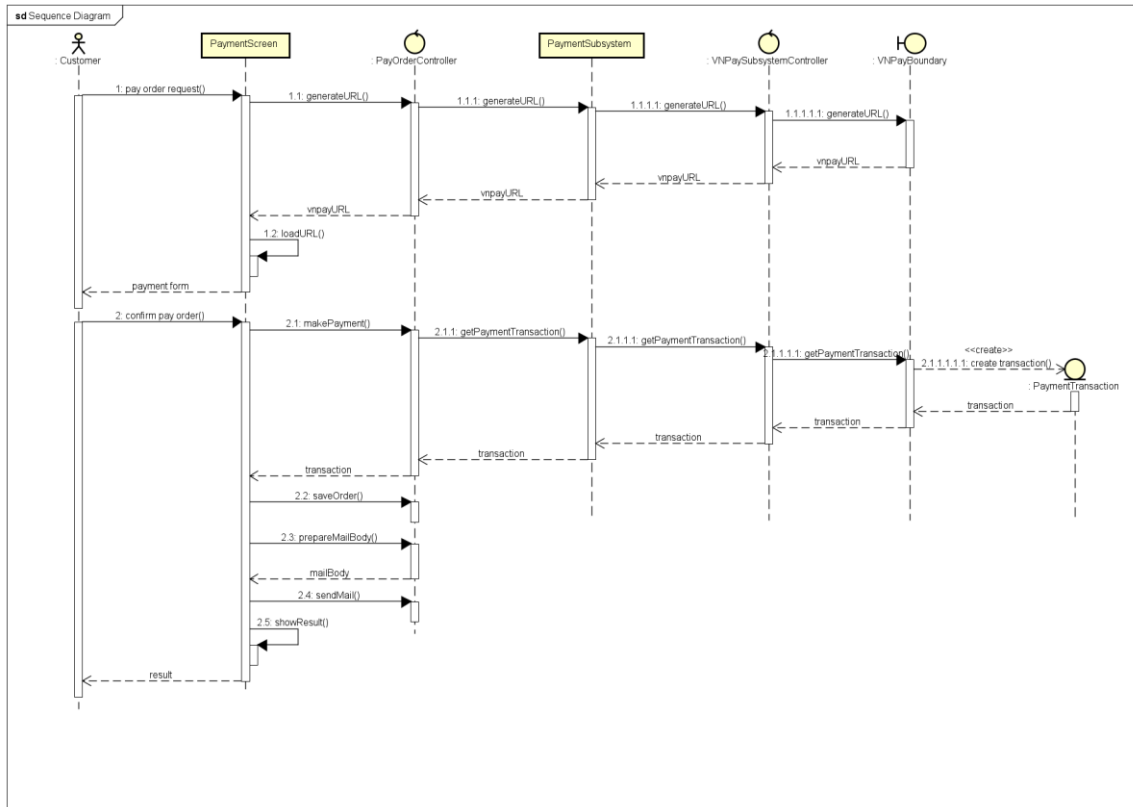
3.2.3 Place Order sequence diagram



3.2.4 Place Rush Order sequence diagram

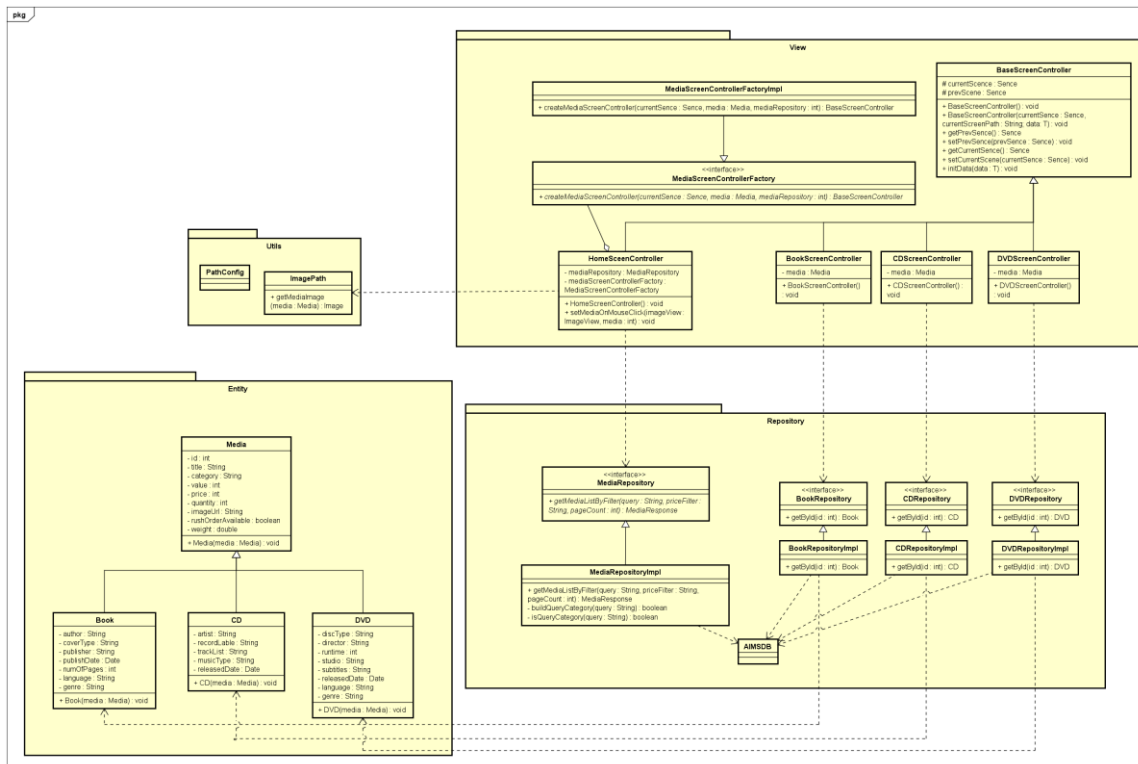


3.2.5 Pay Order sequence diagram

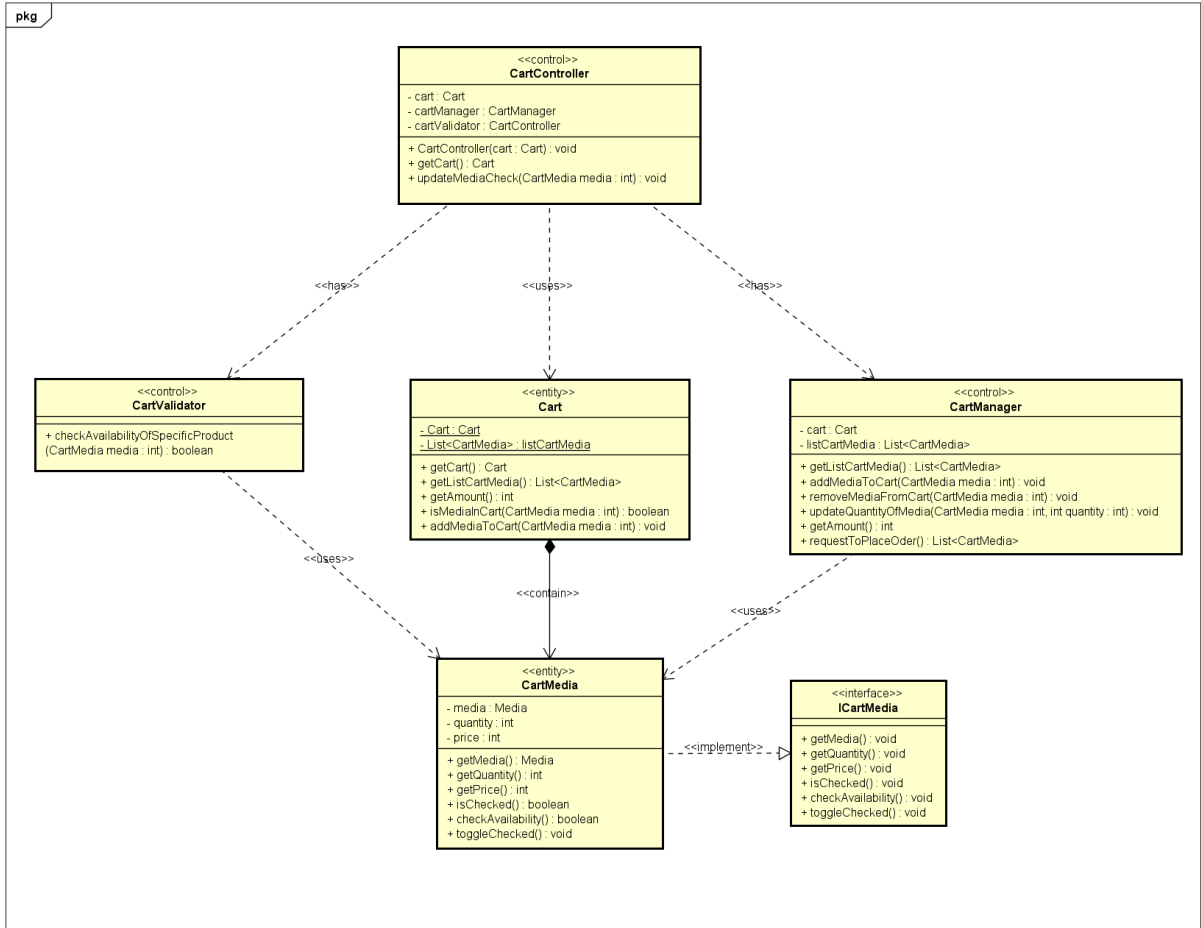


3.3 Analysis Class Diagrams

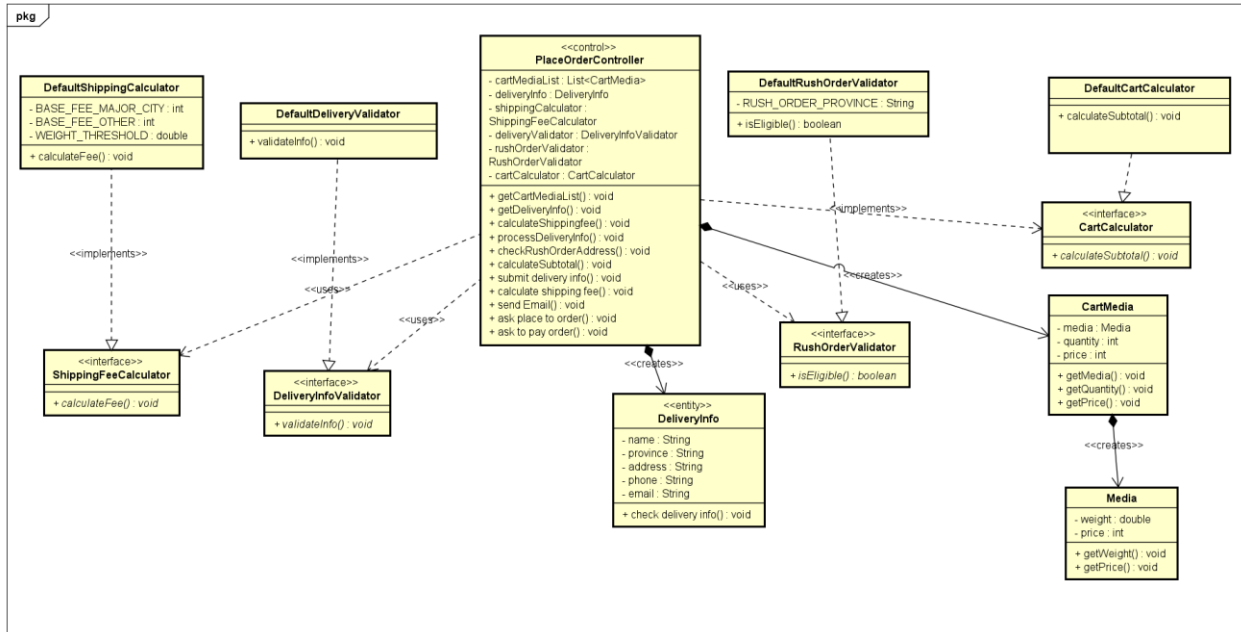
3.3.1 Browse product class diagram



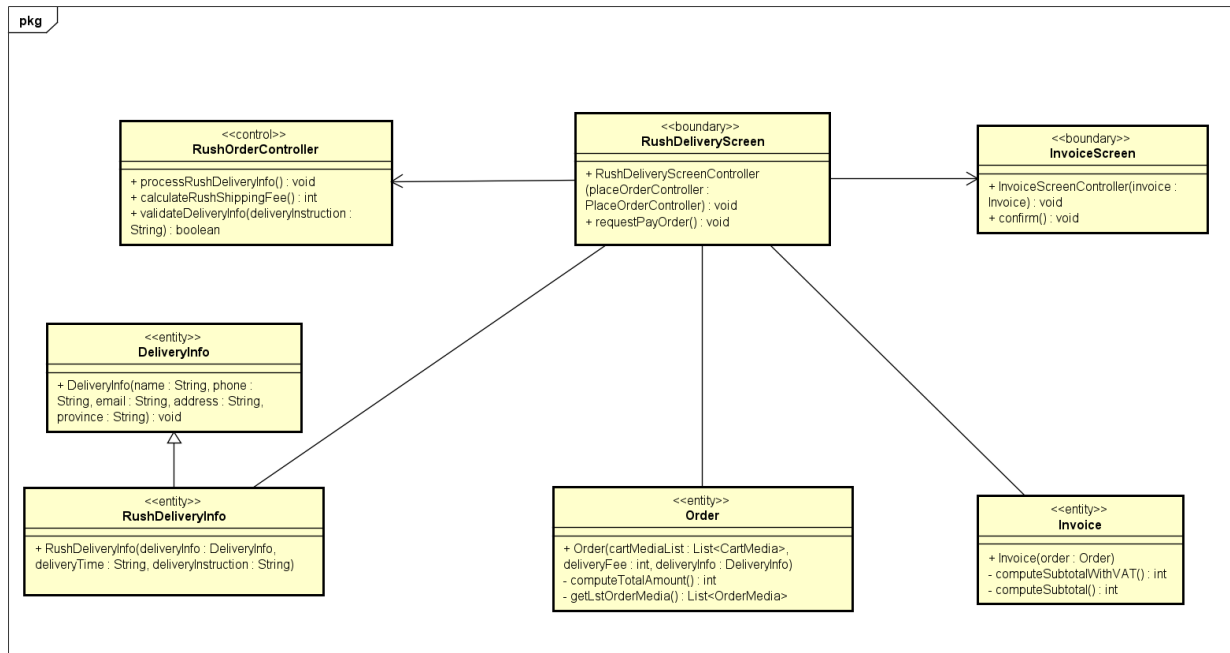
3.3.2 Manage cart class diagram



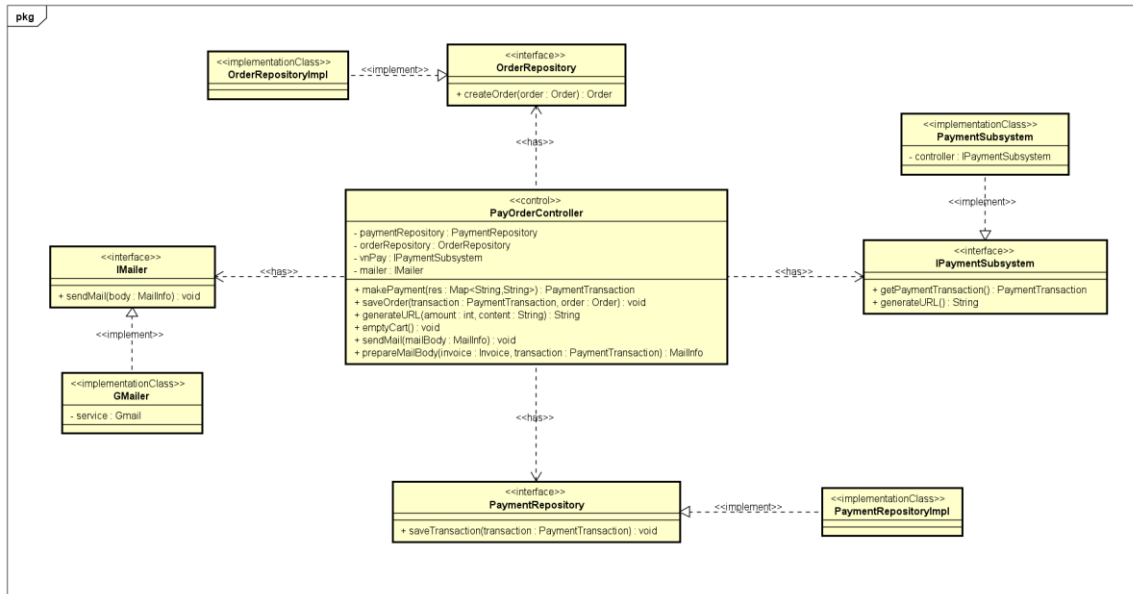
3.3.3 Place Order class diagram



3.3.4 Place Rush Order class diagram



3.3.5 Pay Order class diagram



3.4 Unified Analysis Class Diagram

3.5 Security Software Architecture

<Describe the software components and configuration supporting the security and privacy of the system. Specify the architecture for (1) authentication to validate user identity before allowing access to the system;(2) authorization of users to perform functional activity once logged into the system, (3) encryption protocol to support the business risks and the nature of information, and (4) logging and auditing design, if required.>

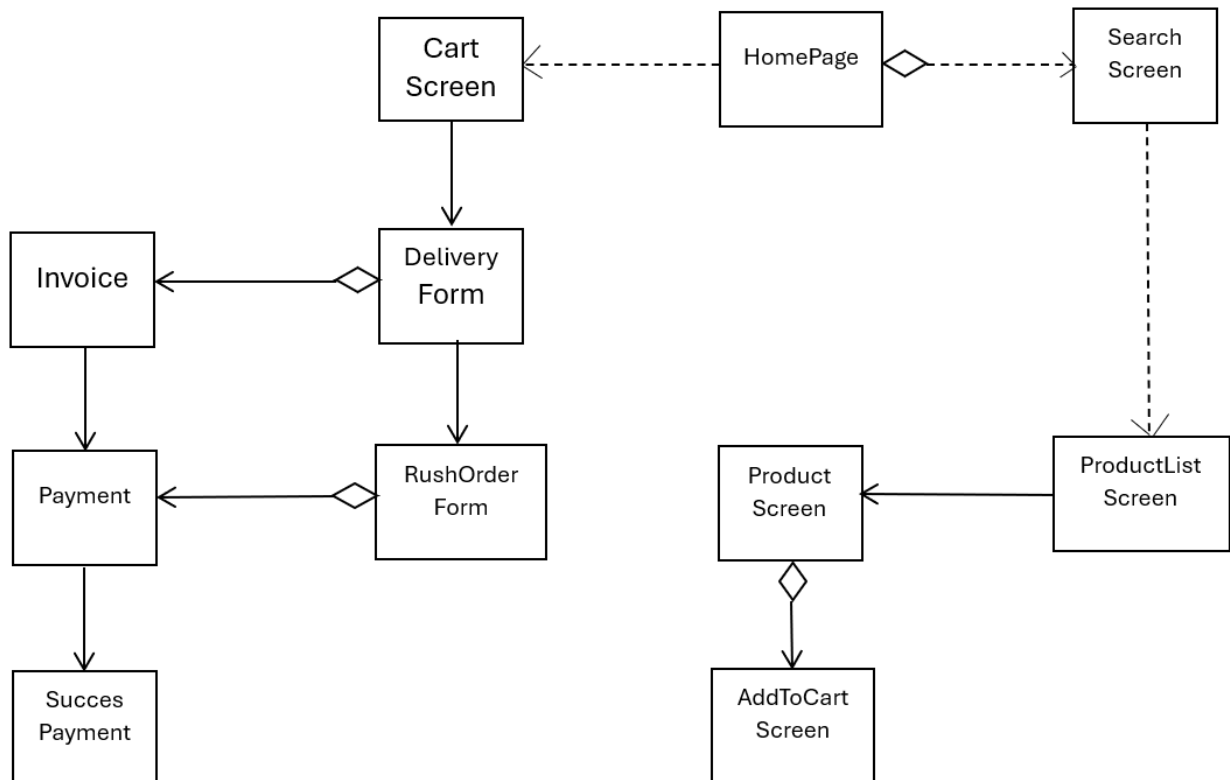
4 Detailed Design

4.1 User Interface Design

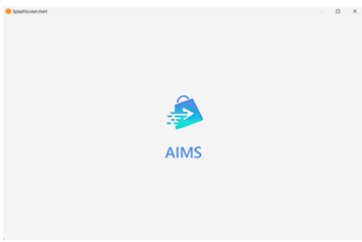
4.1.1 Screen Configuration Standardization

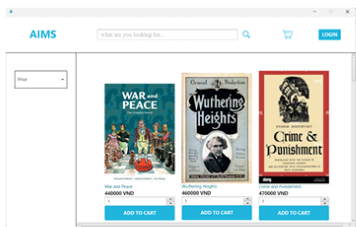
- Kích thước màn hình: 1024x768 (hoặc lớn hơn).
- Font chữ: Arial, kích thước 14px.
- Màu sắc: (Cần định nghĩa bảng màu cho hệ thống).
- Layout: Sử dụng các layout container của JavaFX như HBox, VBox, FlowPane, AnchorPane để bố trí các thành phần giao diện.
- Các component: Sử dụng các component chuẩn của JavaFX như Label, Button, TextField, ImageView, ComboBox, Spinner, ScrollPane, TableView,...


4.1.2 Screen Transition Diagrams



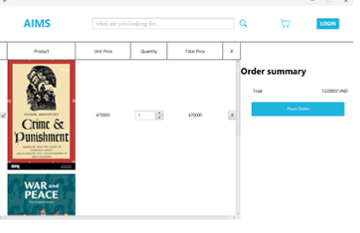
4.1.3 Screen Specifications


<i>AIMS Software</i>		<i>Date of creation</i>	<i>Approved by</i>	<i>Reviewed by</i>	<i>Person in charge</i>
Screen specification	splash screen	04/01/2025			<i>Vũ Việt Phương</i>
		<i>Control</i>	<i>Operation</i>	<i>Function</i>	
		Display start screen	Initial		


<i>AIMS Software</i>		<i>Date of creation</i>	<i>Approved by</i>	<i>Reviewed by</i>	<i>Person in charge</i>
Screen specification	home screen	04/01/2025			<i>Vũ Việt Phương</i>
		<i>Control</i>	<i>Operation</i>	<i>Function</i>	
		<i>price choicebox</i>	choose a sort type	sort media by sort type	
		<i>navigation bar</i>	click/enter	navigate to different screens	
		<i>media image</i>	click	display media details	
		<i>quantity spinner</i>	click	change quantity	
		<i>add to cart button</i>	click	add media to cart	

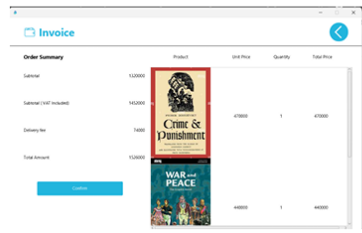
<i>AIMS Software</i>		<i>Date of creation</i>	<i>Approved by</i>	<i>Reviewed by</i>	<i>Person in charge</i>
Screen specification	navigation bar	04/01/2025			Vũ Việt Phương
		Control	Operation	Function	
		<i>AIMS button</i>	click	go to homepage screen	
		<i>search bar</i>	enter	fill the search bar	
		<i>search icon</i>	click	find the medias according the search bar	
		<i>cart button</i>	click	go to cart screen	
		<i>log in</i>	click	go to login screen	

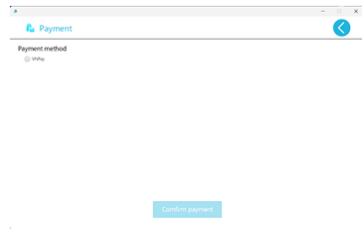
<i>AIMS Software</i>		<i>Date of creation</i>	<i>Approved by</i>	<i>Reviewed by</i>	<i>Person in charge</i>
Screen specification	cart screen	04/01/2025			Nguyễn Minh Phương

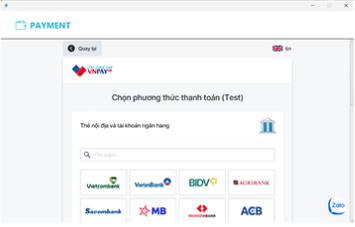
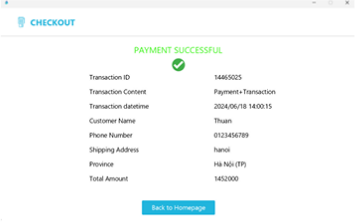

	Control	Operation	Function
	<i>select box</i>	click	select or unselect media want to place order
	<i>quantity spinner</i>	click	change quantity of media in cart
	<i>delete button</i>	click	remove media from cart
	<i>place order button</i>	click	go to delivery screen, to continue place order process

<i>AIMS Software</i>		<i>Date of creation</i>	<i>Approved by</i>	<i>Reviewed by</i>	<i>Person in charge</i>
Screen specification	delivery information screen	04/01/2025			Nguyễn Cảnh Phước
		<i>Control</i>	<i>Operation</i>	<i>Function</i>	
		<i>back button</i>	click	go to previous screen	
		<i>name field</i>	enter	fill the name field	
		<i>phone field</i>	enter	fill the phone field	
		<i>province choicebox</i>	enter	choose a province	
		<i>address field</i>	enter	fill the address field	
		<i>email field</i>	enter	fill the email field	
		<i>rush order select box</i>	click	select or unselect rush order delivery	
		<i>confirm button</i>	click	go to next screen to continue place order	

<i>AIMS Software</i>		<i>Date of creation</i>	<i>Approved by</i>	<i>Reviewed by</i>	<i>Person in charge</i>
Screen specification	rush order screen	04/01/2025			<i>Giang Trung Quân</i>
		<i>Control</i>	<i>Operation</i>	<i>Function</i>	
		<i>time choicebox</i>	click	choose a time	
		<i>delivery instruction field</i>	enter	fill the instruction	
		<i>confirm button</i>	click	go to next screen to continue place order process	
		<i>back button</i>	click	go to the previous screen	

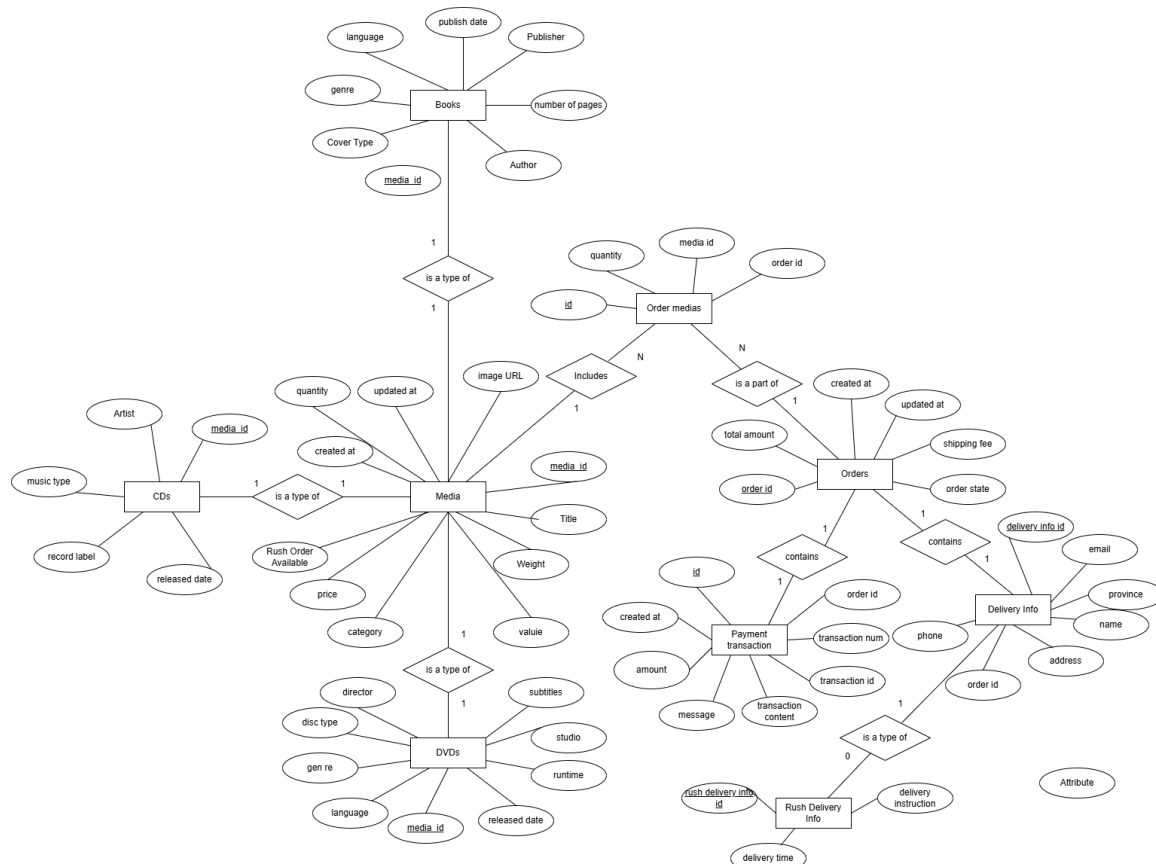
AIMS Software		Date of creation	Approved by	Reviewed by	Person in charge
Screen specification		04/01/2025			Bùi Trung Quân
		Control	Operation	Function	
		back button	click	go to the previous screen	
		confirm button	click	go to payment method screen	

AIMS Software		Date of creation	Approved by	Reviewed by	Person in charge
Screen specification	payment method screen	04/01/2025			Bùi Trung Quân
		Control	Operation	Function	
		payment method radio	click	choose a payment method	
		back button	click	go to the previous screen	
		confirm button	click	go to VNPay boundary	

<i>AIMS Software</i>		<i>Date of creation</i>	<i>Approved by</i>	<i>Reviewed by</i>	<i>Person in charge</i>
Screen specification	VNPay boundaray	04/01/2025			Bùi Trung Quân
		<i>Control</i>	<i>Operation</i>	<i>Function</i>	
<i>AIMS Software</i>		<i>Date of creation</i>	<i>Approved by</i>	<i>Reviewed by</i>	<i>Person in charge</i>
Screen specification	success payment screen	04/01/2025			Bùi Trung Quân
		<i>Control</i>	<i>Operation</i>	<i>Function</i>	
		<i>back to homepage button</i>	click	go to homepage screen	
<i>AIMS Software</i>		<i>Date of creation</i>	<i>Approved by</i>	<i>Reviewed by</i>	<i>Person in charge</i>
Screen specification	media detail screen	04/01/2025			Vũ Việt Phương
		<i>Control</i>	<i>Operation</i>	<i>Function</i>	
		<i>quantity spinner</i>	click	change desired cart's media quantity to add to cart	
		<i>add to cart button</i>	click	add to cart	

4.2 Data Modeling

4.2.1 Conceptual Data Modeling



4.2.2 Database Design

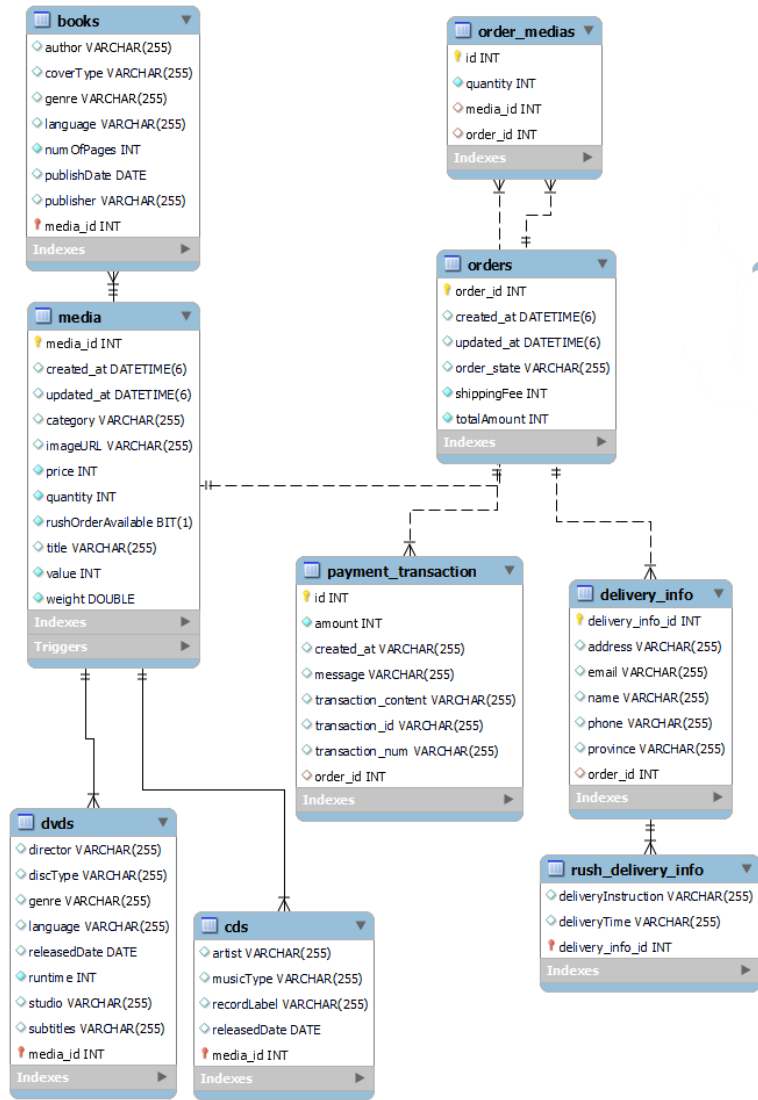
4.2.2.1 Database Management System

DBMS: MySQL

Lý do:

- MySQL là một hệ quản trị cơ sở dữ liệu quan hệ (RDBMS) mã nguồn mở, phổ biến và được sử dụng rộng rãi.
- MySQL có hiệu năng tốt, ổn định và dễ sử dụng.
- MySQL có cộng đồng hỗ trợ đông đảo và nhiều tài liệu hướng dẫn.

4.2.2.2 Database Diagram



4.2.2.3 Database Detail Design

Table 1. Media

#	PK	FK	Column name	Data type	Default value	Mandatory	Description
1	x		media_id	integer		yes	

2			category	varchar		yes	type of media, could be book, CD, DVD
---	--	--	----------	---------	--	-----	---

3			created_at	Datetime	current date	yes	creation date
4			updated_at	Datetime	current date	yes	update date
5			imageURL	String		no	path to image
6			price	integer		yes	price of the media
7			value	integer		yes	value of the media
8			quantity	integer		yes	number of media in stock
9			Rush Order available	boolean		yes	
10			title	varchar		yes	title of media
11			weight	double		yes	weight of the media

Triggers:

- media_create: On insert into table media, set created_at and updated_at to now.

Table 2. Book

#	PK	FK	Column name	Data type	Default value	Mandatory	Description
1	x	x	media_id	integer		yes	
2			author	varchar		yes	author of the book

3			coverType	varchar		yes	book cover type
---	--	--	-----------	---------	--	-----	-----------------

4			genre	varchar		yes	book genre
5			language	varchar		yes	book language
6			numOfPages	integer		yes	number of pages
7			publisher	varchar		yes	book publisher
8			publishDate	date		yes	book publish date

Table 3. CD

#	PK	FK	Column name	Data type	Default value	Mandatory	Description
1	x	x	media_id	integer		yes	
2			artist	varchar		yes	artist of the CD
3			musicType	varchar		yes	CD music type
4			record label	varchar		yes	CD record label
5			releasedDate	date		yes	CD released date

Table 4. DVD

#	PK	FK	Column name	Data type	Default value	Mandatory	Description
1	x	x	media_id	integer		yes	
2			director	varchar		yes	director of the DVD
3			discType	varchar		yes	DVD disc type
4			genre	varchar		yes	DVD genre
5			language	varchar		yes	DVD language
6			runtime	integer		yes	runtime of DVD
7			studio	varchar		yes	DVD studio
8			subtitles	varchar		yes	DVD subtitles
9			releasedDate	date		yes	DVD release date

Table 5. Order Media

#	PK	FK	Column name	Data type	Default value	Mandatory	Description
---	----	----	-------------	-----------	---------------	-----------	-------------

1	x		id	integer		yes	
2		x	media_id	integer		yes	

3		x	order_id	integer		yes	
4			quantity	integer		yes	number of one media in order

Table 6. Order

#	PK	FK	Column name	Data type	Default value	Mandatory	Description
1	x		order_id	integer		yes	
2			created_at	datetime	current date	yes	
3			updated_at	datetime	current date	yes	
4			totalAmount	integer		yes	total amount of the order
5			shippingFee	integer		yes	shipping fee of order
6			orderState	varchar		yes	could be pending, declined, delivering, delivered

Table 7. Delivery Info

#	<i>PK</i>	<i>FK</i>	<i>Column name</i>	<i>Data type</i>	<i>Default value</i>	<i>Mandatory</i>	<i>Description</i>
1	x		delivery_info_id	integer		yes	
2		x	order_id	integer		yes	
3			name	vvarchar		yes	recipient name
4			phone	vvarchar		yes	phone number
5			email	vvarchar		yes	
6			address	vvarchar		yes	delivery address
7			province	vvarchar		yes	

Table 8. Rush delivery info

#	<i>PK</i>	<i>FK</i>	<i>Column name</i>	<i>Data type</i>	<i>Default value</i>	<i>Mandatory</i>	<i>Description</i>
1	x	x	delivery_info_id	integer		yes	

2			delivery time	vvarchar		yes	delivery time frame
---	--	--	---------------	----------	--	-----	------------------------

3			delivery instruction	vvarchar		yes	
---	--	--	----------------------	----------	--	-----	--

Table 9. Payment Transaction

#	PK	FK	Column name	Data type	Default value	Mandatory	Description
1	x		id	integer		yes	
2		x	order_id	integer		yes	
3			created_at	vvarchar		yes	
4			amount	integer		yes	amount in transaction
5			transaction_number	vvarchar		yes	
6			transaction_content	vvarchar		no	
7			message	vvarchar		no	
8			transaction_id	vvarchar'		yes	

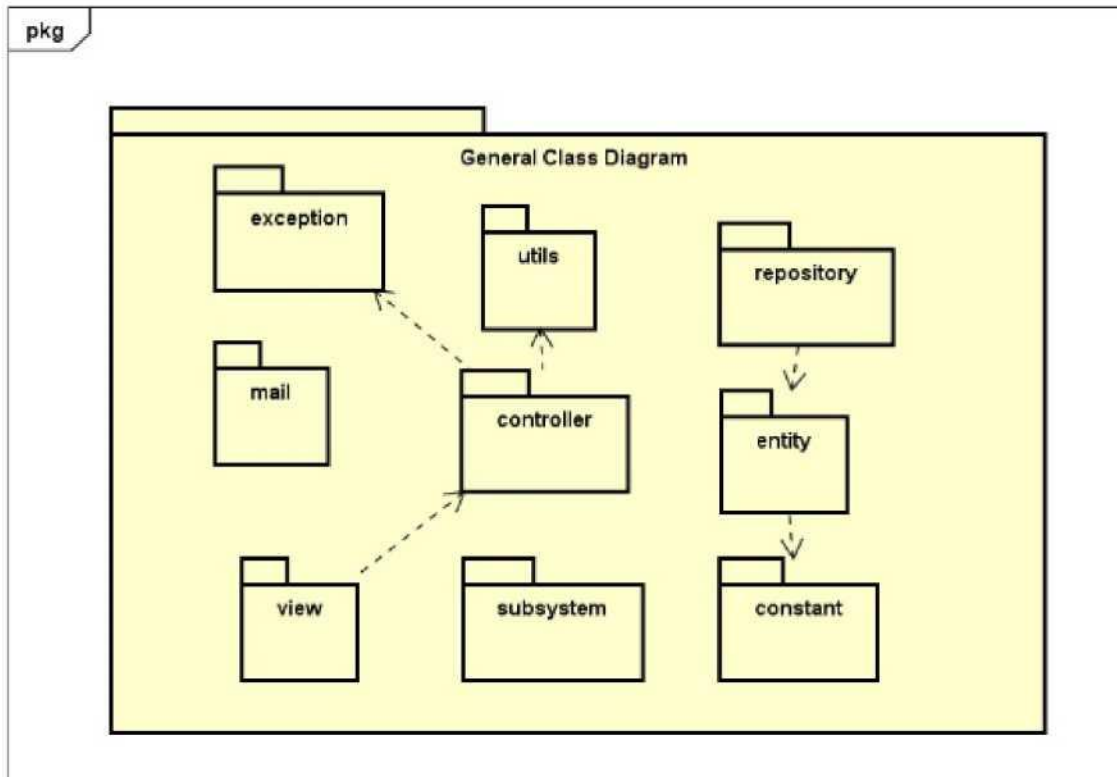
4.3 Non-Database Management System Files

<Provide the detailed description of all non-DBMS files if any and include a narrative description of the usage of each file that identifies if the file is used for input, output, or both, and if the file is a temporary file. Also provide an indication of which modules read and write the file and include file structures (refer to the data dictionary). As appropriate, the file structure information should include the following:

- *Record structures, record keys or indexes, and data elements referenced within the records*
- *Record length (fixed or maximum variable length) and blocking factors*
- *Access method (e.g., index sequential, virtual sequential, random access, etc.)*
- *Estimate of the file size or volume of data within the file, including overhead resulting from file access methods*
- *Definition of the update frequency of the file (If the file is part of an online transaction-based system, provide the estimated number of transactions per unit of time, and the statistical mean, mode, and distribution of those transactions.)*
- *Backup and recovery specifications>*

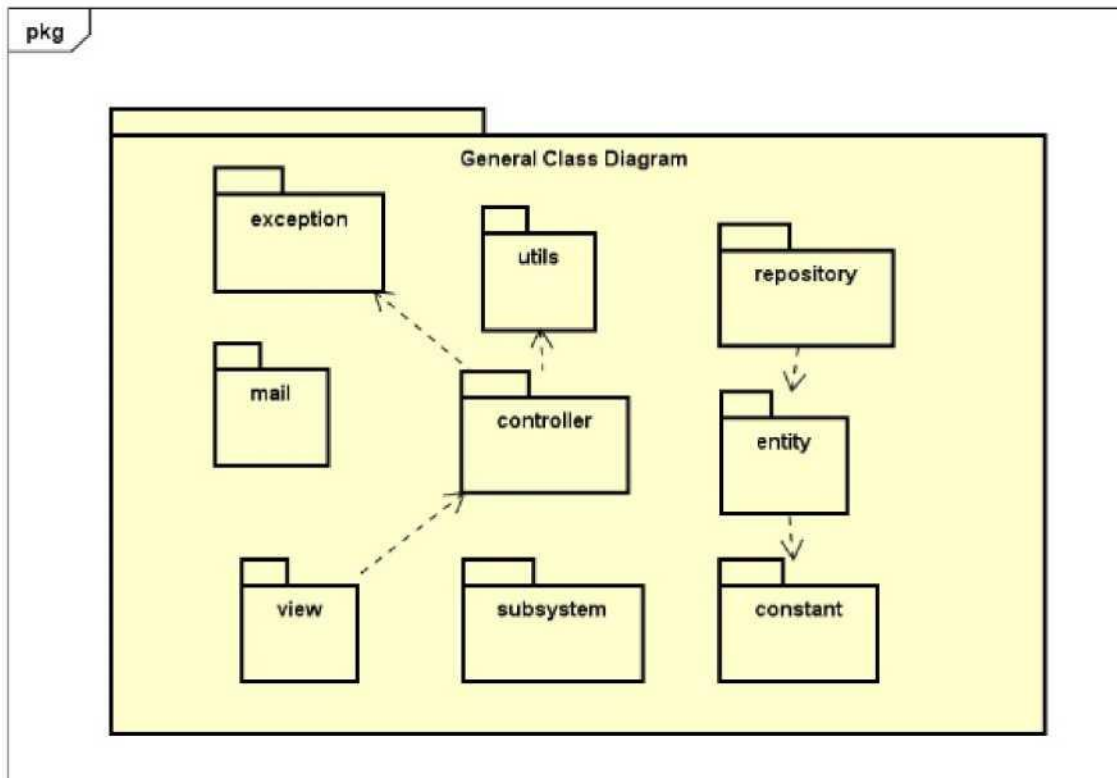
4.4 Class Design

4.4.1 General Class Diagram

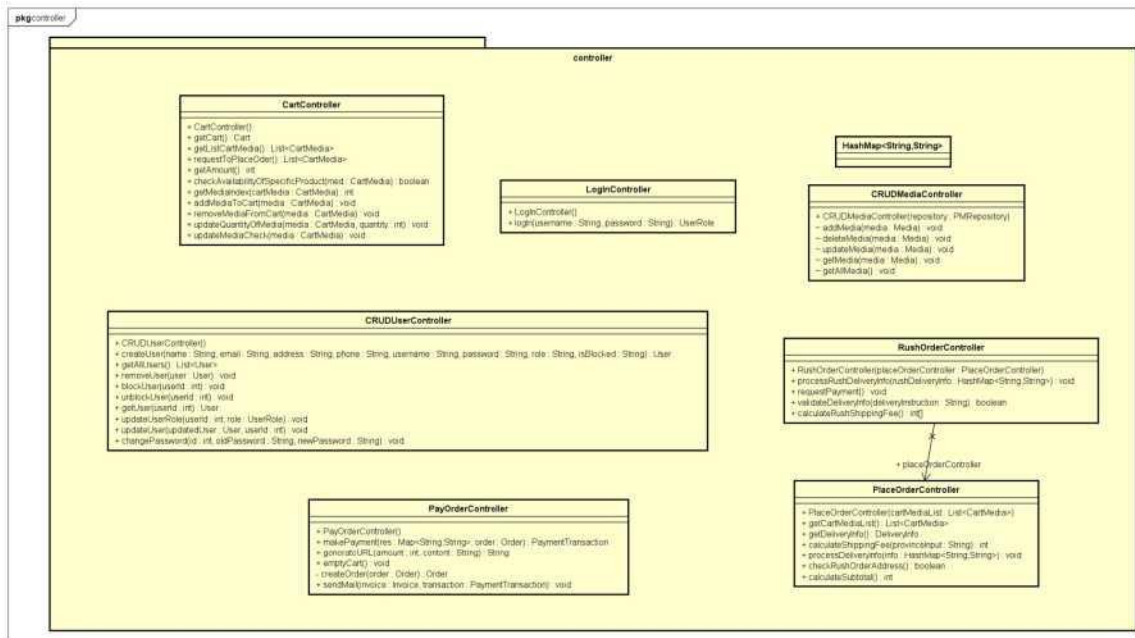


4.4.2 Class Diagrams

4.4.2.1 Class Diagram for Package Entity

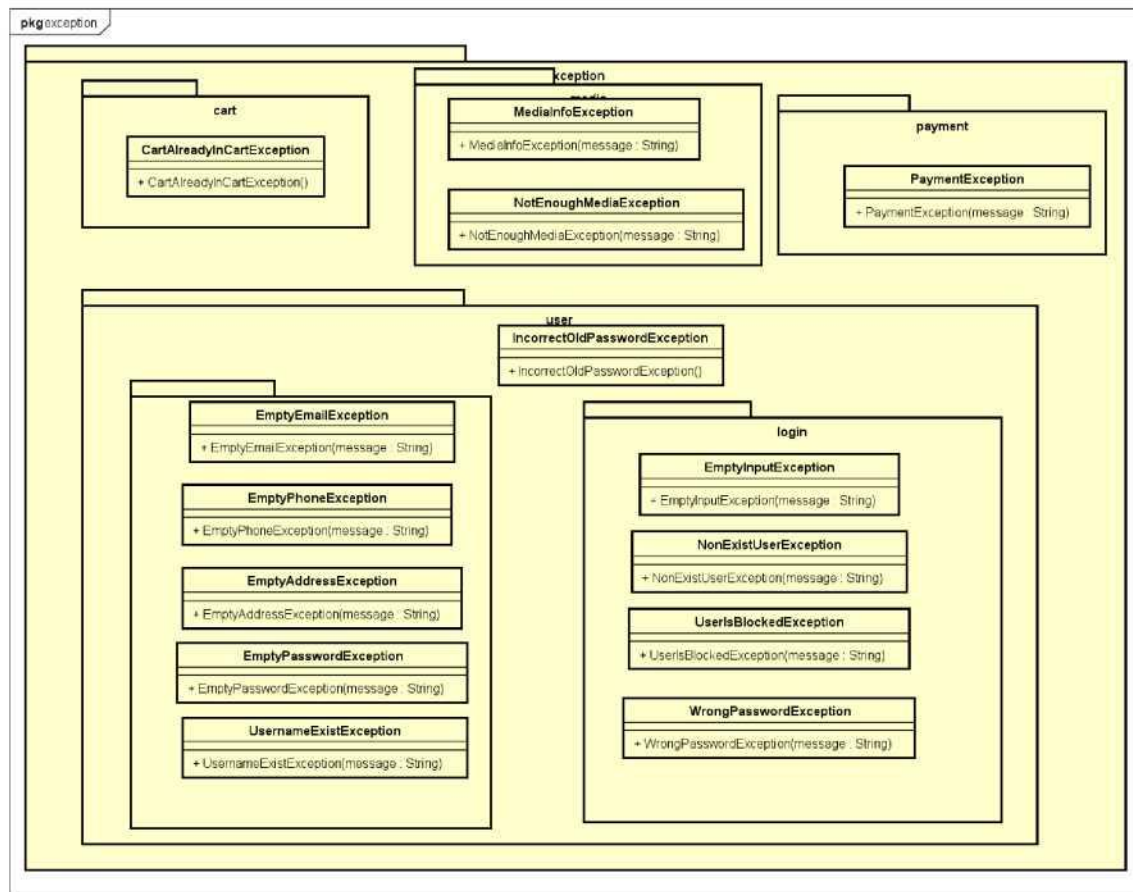


4.4.2.2 Class Diagram for Package Controller

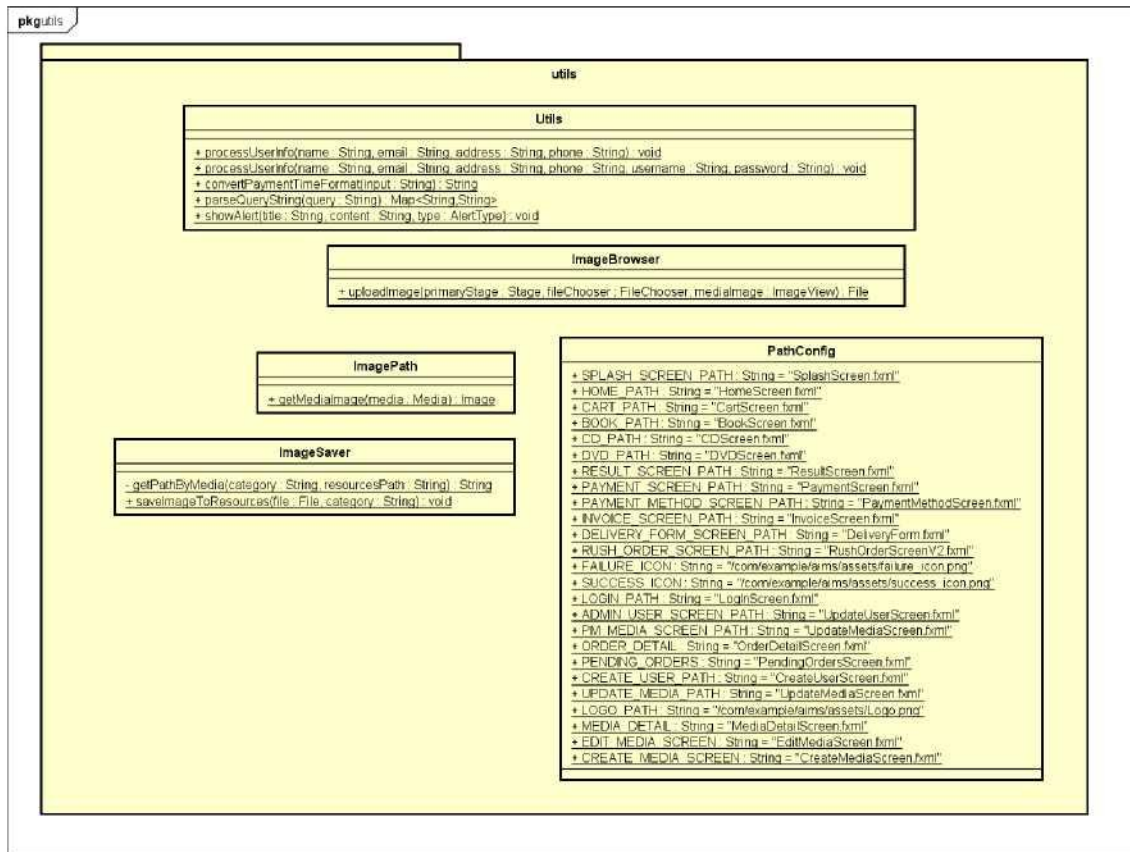


4.4.1.3 Class Diagram for Package Repository

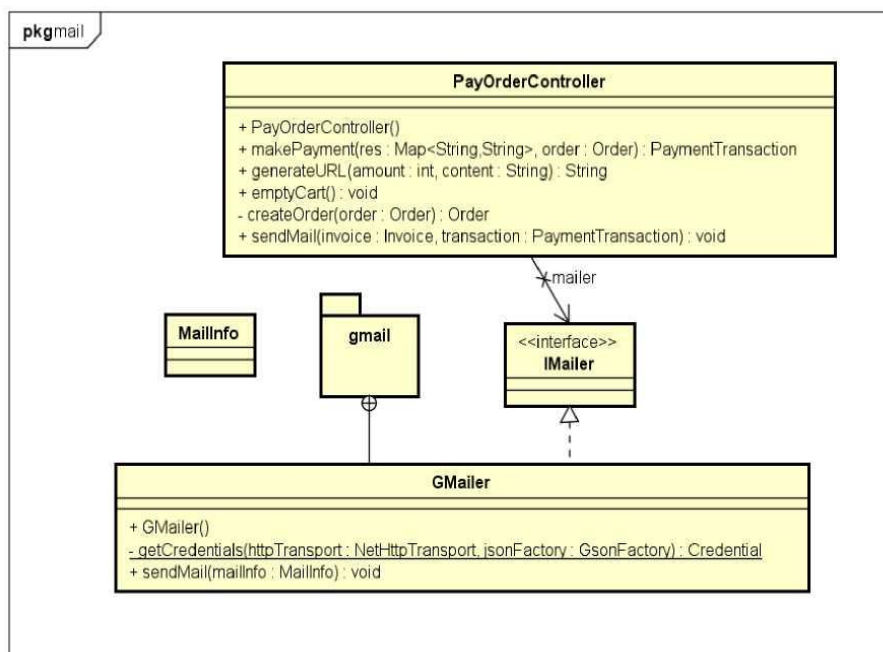
4.4.1.5 Class Diagram for Package Exception



4.4.1.6 Class Diagram for Package Utils



4.4.1.9 Class Diagram for Package Email



5 Design Considerations

5.1 Goals and Guidelines

Mục tiêu:

- Hệ thống hoạt động ổn định, đáp ứng được nhu cầu của người dùng.
- Giao diện người dùng thân thiện, dễ sử dụng.
- Code dễ đọc, dễ bảo trì và nâng cấp.

Hướng dẫn:

- Tuân thủ các nguyên tắc SOLID.
- Sử dụng các design pattern phù hợp.
- Viết code rõ ràng, có comment đầy đủ.
- Kiểm thử code thường xuyên.

5.2 Architectural Strategies

Sử dụng kiến trúc **MVC** để tách biệt rõ ràng các thành phần của hệ thống.

Sử dụng **Factory Pattern** để tạo các đối tượng *MediaRepository* và **ScreenController*.

Sử dụng **Dependency Injection** để giảm coupling giữa các lớp.

Sử dụng **Repository Pattern** để truy cập cơ sở dữ liệu.

Sử dụng **Singleton Pattern** cho lớp *Cart*.

Sử dụng **Strategy**

Pattern cho *AddMediaStrategy*, *RemoveMediaStrategy* và *CheckAvailabilityStrategy* trong *CartController*.

Sử dụng **Service Layer Pattern** để định nghĩa các dịch vụ của hệ thống.

Ngôn ngữ lập trình: **Java**.

Framework: **JavaFX** cho giao diện người dùng.

Database: **MySQL**.

Cổng thanh toán: **VNPay**.

5.3 *Coupling and Cohesion*

Coupling: Cố gắng giảm thiểu coupling giữa các lớp bằng cách sử dụng Dependency Injection, Factory Pattern và các interface. Hiện tại, coupling giữa các lớp ở mức chấp nhận được.

Data Coupling: HomeScreenController và *ScreenController với CartController.

Control Coupling: HomeScreenController với Factory và các *ScreenController.

Cohesion: Cố gắng đảm bảo các lớp có tính gắn kết cao, mỗi lớp chỉ thực hiện một chức năng cụ thể.

Giải pháp: Tách các class lớn thành các class nhỏ hơn, mỗi class có một trách nhiệm duy nhất.

5.4 *Design Principles*

SOLID:

Single Responsibility Principle (SRP): Mỗi lớp chỉ có một trách nhiệm duy nhất. (Ví dụ: Media chỉ lưu trữ thông tin sản phẩm, Cart chỉ quản lý giỏ hàng).

Open/Closed Principle (OCP): Dễ dàng mở rộng chức năng của hệ thống bằng cách thêm các lớp mới mà không cần sửa đổi các lớp hiện có. (Ví dụ: Thêm loại media mới bằng cách tạo thêm lớp MediaRepository và cập nhật MediaRepositoryFactory).

Liskov Substitution Principle (LSP): Các lớp con có thể thay thế lớp cha mà không làm thay đổi tính đúng đắn của chương trình. (Ví dụ: Book, CD, DVD có thể thay thế cho Media).

Interface Segregation Principle (ISP): Tách các interface lớn thành các interface nhỏ hơn, phù hợp với từng nhóm chức năng. (Ví dụ: Tách MediaRepository thành các interface nhỏ hơn như BookRepository, CDRepository).

Dependency Inversion Principle (DIP): Các module cấp cao không nên phụ thuộc vào các module cấp thấp. Cả hai nên phụ thuộc vào abstraction. (Ví dụ: HomeScreenController phụ thuộc vào interface MediaRepository thay vì MediaRepositoryImpl).

5.5 *Design Patterns*

Factory Pattern: MediaRepositoryFactory, MediaScreenControllerFactory

Dependency Injection: Inject MediaRepository, CartController, MediaScreenControllerFactory vào các lớp cần sử dụng.

Repository Pattern: MediaRepository, BookRepository, CDRepository, DVDRepository, LPRepository

Singleton Pattern: Cart

Strategy Pattern: AddMediaStrategy, RemoveMediaStrategy, CheckAvailabilityStrategy, ShippingFeeCalculator, DeliveryInfoValidator, RushOrderValidator, CartCalculator