HANOI UNIVERSITY OF SCIENCE AND TECHNOLOGY School of Information and Communications Technology

Final Report Version 1.1

AIMS: An Internet Media Store Software Design and Construction

> Group 02 Trần Xuân Bách 20204634 Nguyễn Ngọc Quỳnh Anh 20204631 Nguyễn Vũ Thục Anh 20204632 Vũ Đức Anh 20193985

Contents

1.	,	Team's members contribution	3
2.	1	Use Case Diagrams	3
	2.1.	Use Case Diagram	3
	2.2.	Businesss processes	4
	2.3.	Case Specification	5
	2.3.1	1. UC001: View Invoice	5
	2.3.2	2. UC002: Refund	5
	2.3.3	3. UC003: Search Products	6
	2.3.4	4. UC004 Filter Products	8
	2.3.5	5. UC005 Detail Products	9
3.	1	Use Case Analysis	11
	3.1.	Use Case "Refund"	11
	3.1.1	1. Sequence Diagram for UC "Refund"	11
	3.1.2	2. Analysis Class Diagram for UC "Refund"	11
	3.2.	Use Case "Search Products"	12
	3.2.1	1. Sequence Diagram for UC "Search Products"	12
	3.2.2	2. Analysis Class Diagram for UC "Search Products"	12
	3.3.	Use Case "Filter Products"	13
	3.3.1	1. Sequence Diagram for UC "Filter Products"	13
	3.3.2	2. Analysis Class Diagram for UC "Filter Products"	13
	3.4.	Use Case "Detail Products"	14
	3.4.1	1. Sequence Diagram for UC "Detail Products"	14
	3.4.2	2. Analysis Class Diagram for UC "Detail Products"	14
4.]	Interface Design	15
	4.1.	User Interface Design	15
	4.2.	System Interface Design	20
5.	(Class Design.	23
	5.1.	General Class Diagram	23
	5.2.	Relationship Class Diagram	23
	5.3.	Class Design	24
	5.3.1	1. HomeScreenHandler	24

	5.3.	2.	MediaDetailHandler	25
6.		Da	ta Modeling	26
	6.1.	(Conceptual Data Model	26
	6.2.]	Database Design	26
	<i>5.2.</i>	1. I	Logical Data Model	26
	5.2.	2. I	Physical Data Model	27

1. Team's members contribution

Name	Role	Contribution
Trần Xuân Bách	Team Leader	View list invoice and
		refund function
Nguyễn Ngọc Quỳnh Anh	Member	Search, filter and
		pagination products
Nguyễn Vũ Thục Anh	Member	View detail product
Vũ Đức Anh	Member	View Invoice

2. Use Case Diagrams

2.1. Use Case Diagram

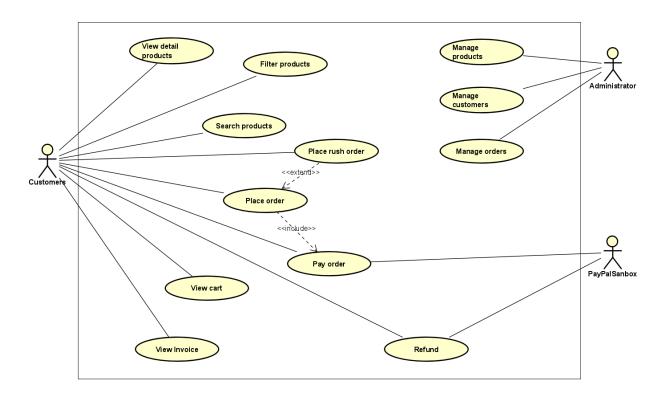


Figure 1: General Use Case Diagram

2.2. Businesss processes

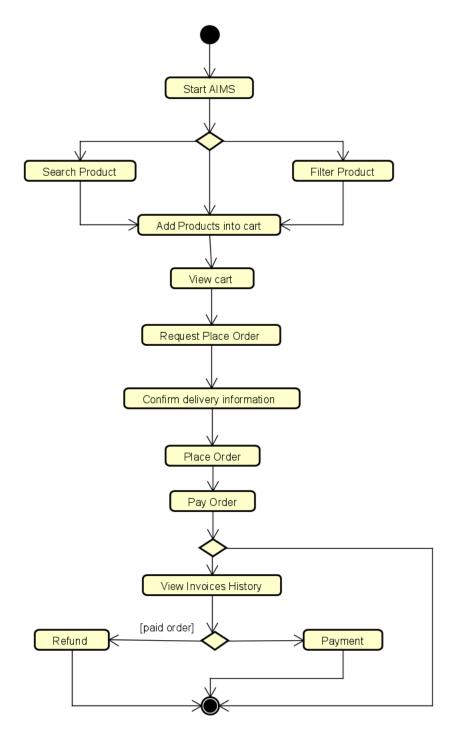


Figure 2: Activity Diagram of customer's usage process

2.3. Case Specification

2.3.1. UC001: View Invoice

2.3.2. UC002: Refund

Use Case "Refund"

1. Use Case code

UC002

2. Brief Description

This use case describes the interaction between Customer and AIMS system when Customer perform order refund.

3. Actors

Customer

4. Preconditions

The customer has viewed invoice

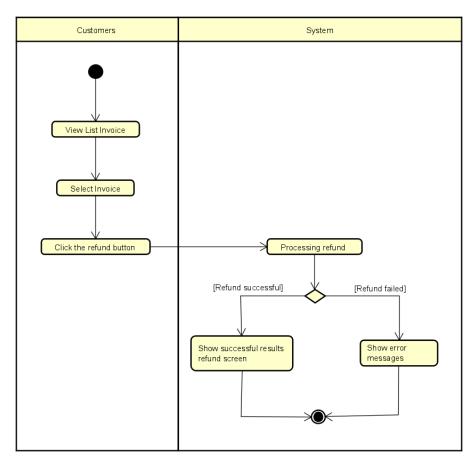
5. Basic Flow of Events

- 1. The system displays list invoice information (UC001)
- 2. The customer selects the invoice that needs a refund
- 3. The system processing the refund
- 4. The system displays successful refund results

6. Alternative Flows

No	Location	Condition	Action	Resume Location
1.	4	If the invoice has expired for a refund	The system displays an error message screen	
2.		If refund processing fails	The system displays the message: Refund failed	

7. Activity Diagram



2.3.3. UC003: Search Products

Use Case "Search Products"

1. Use Case code

UC003

2. Brief Description

This use case describes the interaction between Customer and AIMS system when Customer wishes to search products

3. Actors

Customer

4. Preconditions

The customer has viewed the product list

5. Basic Flow of Events

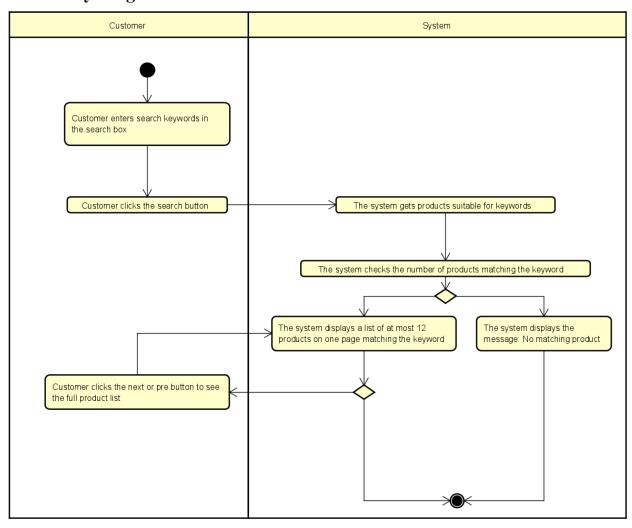
- 1. Customer enters search keywords in the search box
- 2. Customer clicks the search button
- 3. The system gets products suitable for keywords
- 4. The system checks the number of products matching the keyword
- 5. The system displays a list of at most 12 products on one page matching the keyword

6. Alternative Flows

Table 4 -Alternative Flow of Use case "Search Products"

No	Location	Condition	Action	Resume Location
1.	5	If there are no products matching the keyword	■ The system displays the message: There are no matching products	
2.		If there are more than 12 products matching the keyword	 Customer clicks the next or pre button to see the full product list 	5

7. Activity Diagram



2.3.4. UC004 Filter Products

Use Case "Filter Products"

1. Use Case code

UC004

2. Brief Description

This use case describes the interaction between Customer and AIMS system when Customer wishes to filter products

3. Actors

Customer

4. Preconditions

The customer has viewed the product list

5. Basic Flow of Events

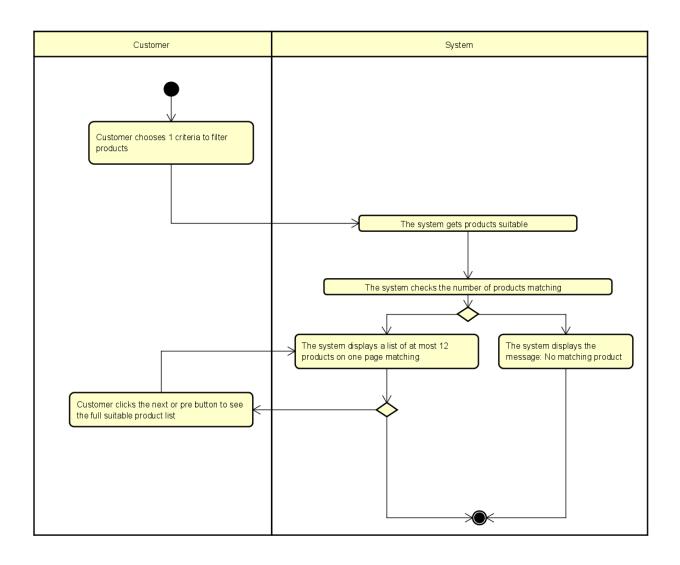
- 1. Customer chooses 1 criteria to filter products
- 2. The system gets products suitable
- 3. The system checks the number of products matching
- 4. The system displays a list of at most 12 products on one page matching

6. Alternative Flows

Table 4 -Alternative Flow of Use case "Filter Products"

No	Location	Condition	Action	Resume Location
1.	5	If there are no products matching	■ The system displays the message: There are no matching products	
2.		If there are more than 12 products matching	 Customer clicks the next or pre button to see the full product list 	4

7. Activity Diagram



2.3.5. UC005 Detail Products

Use Case "Detail Products"

1. Use Case code

UC005

2. Brief Description

This use case describes viewing product details

3. Actors

Customer

4. Preconditions

The customer has viewed detail product

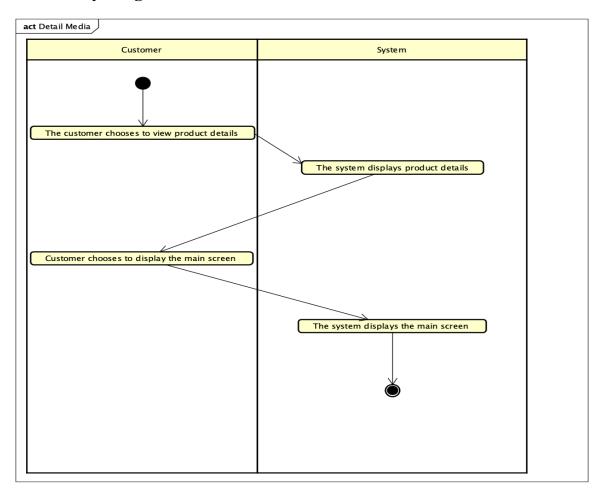
5. Basic Flow of Events

- 1. The customer chooses to view product details
- 2. The system displays product details

- 3. Customer chooses to display the main screen
- 4. The system displays the main screen

6. Alternative Flows

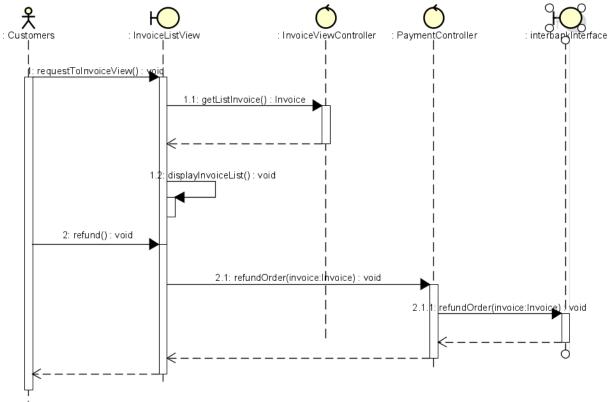
7. Activity Diagram



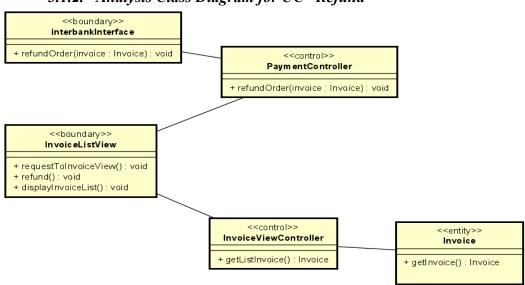
3. Use Case Analysis

3.1. Use Case "Refund"

3.1.1. Sequence Diagram for UC "Refund"

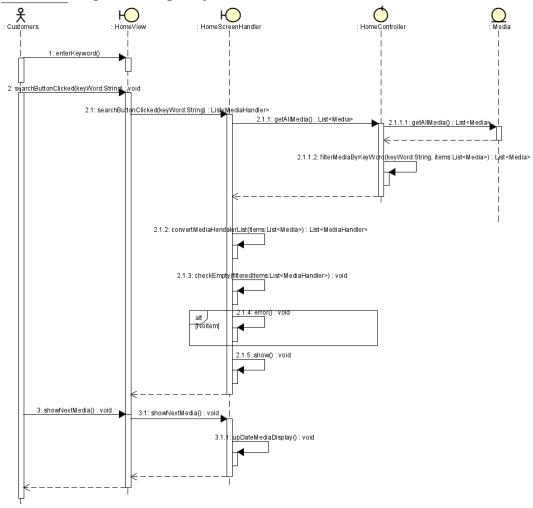


3.1.2. Analysis Class Diagram for UC "Refund"

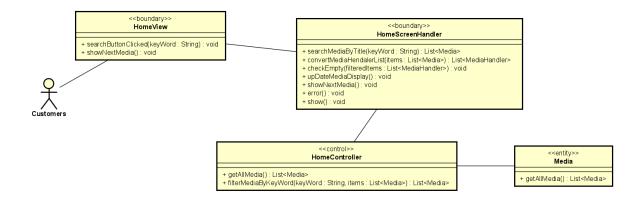


3.2. Use Case "Search Products"

3.2.1. Sequence Diagram for UC "Search Products"

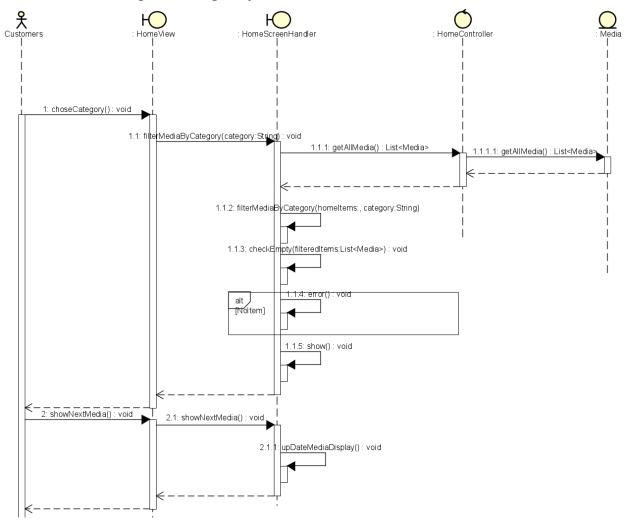


3.2.2. Analysis Class Diagram for UC "Search Products"

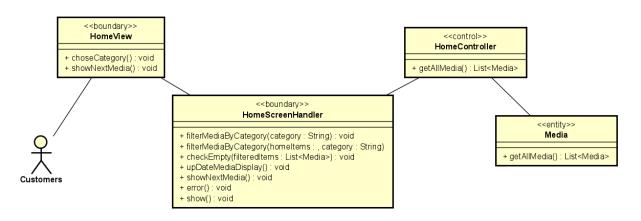


3.3. Use Case "Filter Products"

3.3.1. Sequence Diagram for UC "Filter Products"

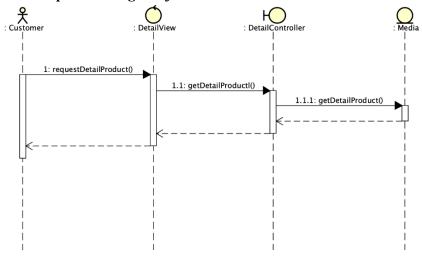


3.3.2. Analysis Class Diagram for UC "Filter Products"

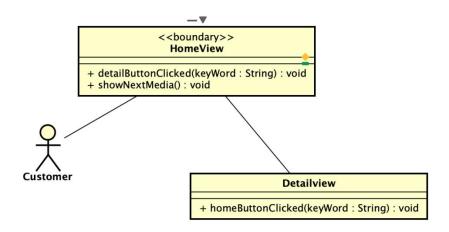


3.4. Use Case "Detail Products"

3.4.1. Sequence Diagram for UC "Detail Products"



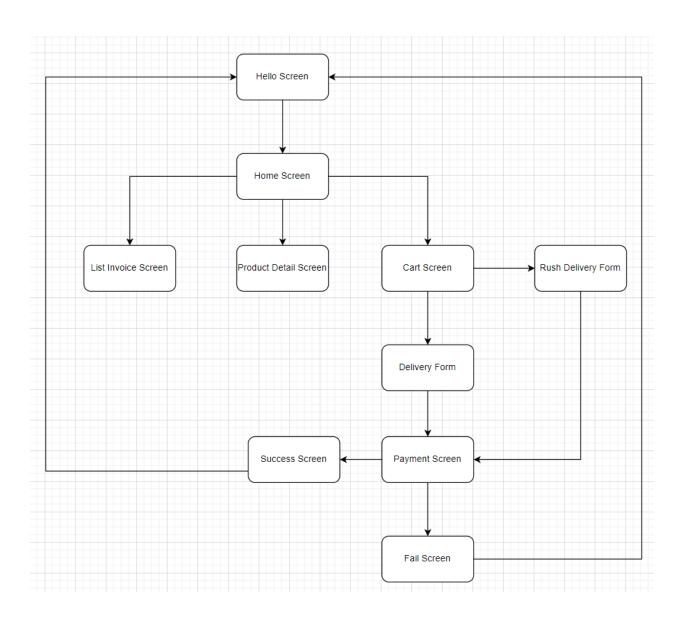
3.4.2. Analysis Class Diagram for UC "Detail Products"



4. Interface Design

4.1. User Interface Design

Screen transition flow



Hello Page



Internet Media Store

Home Page

AIMS Software				Date of creation	Person in o	charge	
Screen specification		ation Home page screen 27.		27/10/2023	Nguyễn Ngọc Quỳnh Anh		
<u>\$</u>			Search ▼		Control	Operator	Function
AIMS					Search Input	Type	Input Search
	book2	book6	book4	dvd12	Area		Keyword
ALE	Avail 12 INDIANANA	Avail 12 GALARY	Avail 12	Avail 17 1 0	Search Button	Click	Search
Remi	Add to Cart	Add to Cert	Add to Cart	Add to Cart			Media with
en-ton	book9	cd7	cd3	book3			input in
ROTHERS	Price 21 d Avail 12	Price 24 d Avail 12	Price 66 d Avail 12	Harry Potter Price 25 d			search area
3	1 0	1 ÷	1 ÷	1 0	Cart Button	Click	Display Cart
	Add to Cert	Add to Cart	Add to Cart	Add to Cart	Cart Button	CHCK	
THE REPORT OF	book10	book12	book1	dvd10			Page
BIG BOOK	Price 73 d KING Avail 12	Price 50 d Smith	Price 79 d Avail 12	Price 75 d Avail 12	Area for	Initial	Display
2006	Add to Cart	1 Add to Cert	1 Add to Cart	DEAD STILL Add to Cart	displaying		Banner
	Add to Cart	Add to Cart	Additional	Fage 1 of 3 Five Next	Banner		
					Area for	Initial	Display
					displaying	IIIIIII	Media
					Media		Micuia
						C1: 1	D: 1
					Next Button	Click	Display next
							page
					Pre Button	Click	Display
							previous
							page

Cart Page

AIMS Software	Date of	Person in charge
	creation	

Screen specification	Home page screen	27/10/2023	Nguyễn N	
			Quỳnh An	h
♦ CART		Control	Operator	Function
book4 Delete dvd12	57.000 d	Area for displaying Price	playing	
Delete	82.000 d	Area for displaying Media	Initial	Display Media
		Place order Button	Click	Display Delivery Form
		Delete Button	Click	Remove product from cart

Delivery Information Form Page

AIMS Softwar	re		Date of creation	Person in	charge
Screen specific	cation	Home page screen 27/10/2023		Nguyễn N Anh	gọc Quỳnh
SHIPPING			Control	Operator	Function
 Name 	(a-zA-Z)		Name Input	Type	Set name
 Phone 	(0-9) 10 digits		area	_	reciever
• City	•		Phone input area	Type	Set phone reciever
• Address	(a-zA-Z)		Province Input area	Choose	Set province reciever
Shipping Instructions	(a-zA-Z)		Address Input area	Туре	Set address reciever
	Confirm delivery		Instruction Input area	Туре	Set instruction reciever
			Submit button	Click	Send form and display Rush Order Form Page or Invoice Page

Invoice Page

AIMS Software	Date of	Person in charge
	creation	

Screen	specification	Home page screen	27/10/2023	Nguyễn Ngọ	c Quỳnh Anh
INVO	ICE		Control	Operator	Function
Name	Qa	dvd6 28 g	Area for	Initial	Display price
Phone City	0987654321 Cao Bầng		displaying Price		
Address	ā		Area for	Initial	Display
Shipping Instructions			displaying Delivery Information		Delivery Information
	Subtotal Shipping Fees Total	30 d 32 d 62 d firm order	Area for displaying Media	Initial	Display Media
			Confirm order Button	Click	Display Payment Page

List Invoice Page

AIMS	IMS Software			Date of creation	Person in charge		
creen	specification	on]	Home page screen	29/11/2023	Trần Xuân Bách	
LIST	INVOICE		•		Control	Operator Function	
STI	Invoice ID	Amount	Status		Delete	Click	DeleteInvoice
21	08U847582R311951J	58	REFUND	Delete	Button		
22	5XJ52192F61819729	58	PAYMENT COMPLETED	Return	Refund	Click RefundO	RefundOrder
23	2D230126LJ414743T	74	REFUND	Delete			RefundOrder
24	5K909474AL4549459	141	REFUND	Delete	Button		
25	2BP640027Y649824X	107	REFUND	Delete			
26	1AB104055U5702716	135	CREATED	Payment			
					Payment	Click	PayOrder
					Buton		1 47 91401
					Duton		
C				13			

Pay Order Page

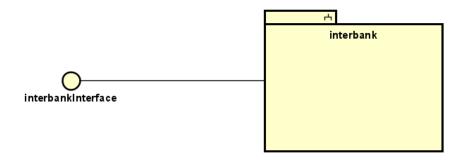
AIMS Software	Date of creation	Person in charge		
Screen specification	Pay Order page screen	29/11/2023	Trần Xuân	n Bách
Payment	_	Control	Operator	Function
Payment method PayPal PayPal https://doi.org/10.1001/10.100	ayment link: ttps://www.sandbox.paypal.com/checkoutnow?t ken=55X72480G50755721 Go to Link	Payment Link Go to link button	Inital	Display paypal sanbox link to pay order Go to link to pay order by Paypal
Confirm payment		Confirm Payment Buton	Click	Confirm payment

Detail Product Page

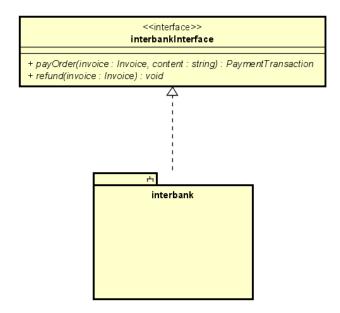
AIMS Software			Date of creation	Person in	charge
Screen specification	Deta	nil product page screen	29/11/2023	Nguyễn V	⁷ ũ Thục Anh
DETAIL			Control	Operator	Function
A manifest park for the second park of the second p	Title Type Category Price Quantity	book2 book story 32 12			

4.2. System Interface Design

3.2.1. Identify subsystems

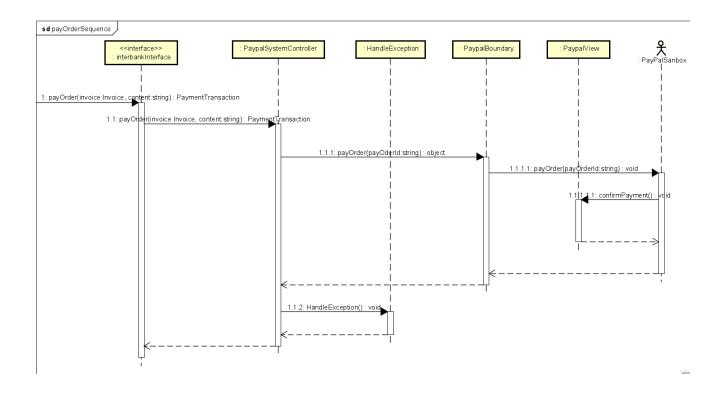


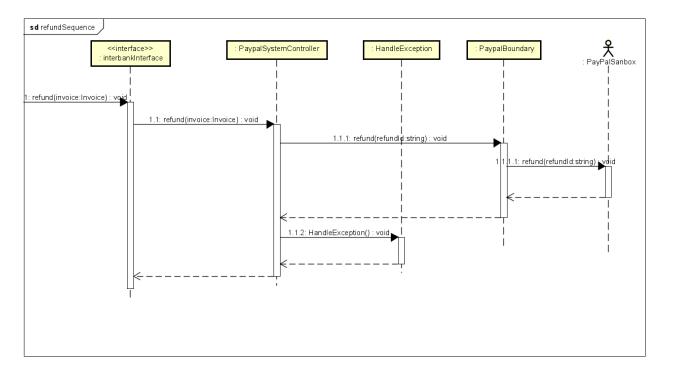
3.2.2. Identify subsystem interface



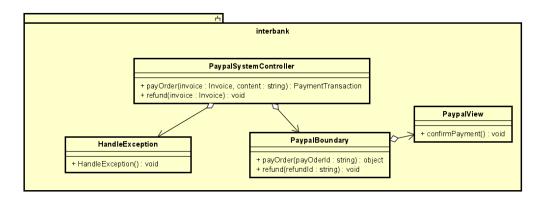
3.2.3. Subsystem design

Distribute subsystem behavior to subsystem elements

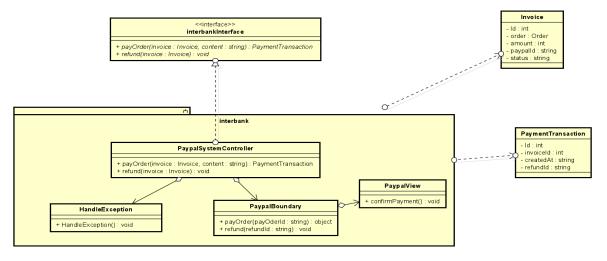




Document subsystem elements

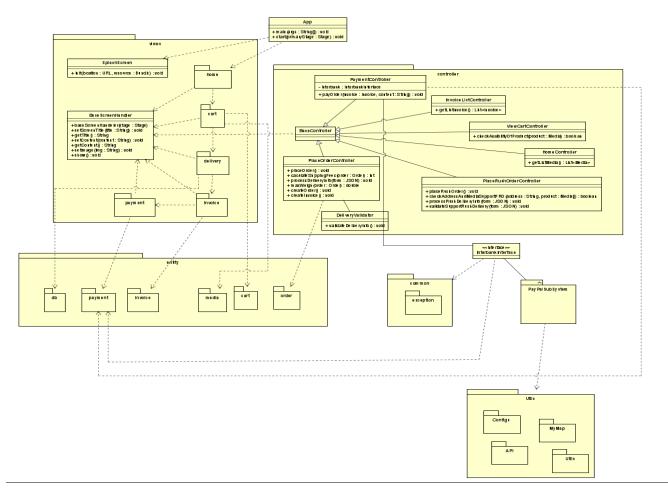


Checkpoints

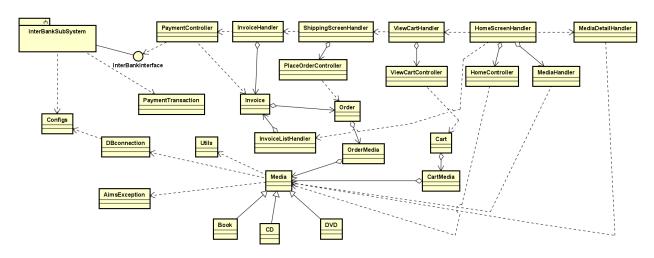


5. Class Design

5.1. General Class Diagram



5.2. Relationship Class Diagram



5.3. Class Design

5.3.1. HomeScreenHandler

HomeScreenHandler
- homeItem : List - displayItems : List - curentPage : int - itemsPerPage : int
+ checkEmpty(filteredItems : List <mediahandler>) : void + upDateMediaDisplay() : void + showNextMedia() : void + showPreviousMedia() : void + addMediaHome() : void + addMenuItem() : void + error() : void</mediahandler>
+ searchButtonClicked(keyWord : String) : List <mediahandler> + convertMediaHendalerList(items : List<media>) : List<mediahandler></mediahandler></media></mediahandler>

Attribute

#	Name	Data type	Default value	Description	
1	homeItems	List	NULL	Holds the media items fetched from the controller.	
2	displayedItems	List	NULL	Holds the currently displayed media items	
3	curentPage	int	0	Tracks the current page number	
3	itemsPerPage	int	12	Represents the number of items to display per page	

Operation

#	Name	Return type	Description (purpose)
1	convertMediaHandlerList	List <mediahandler></mediahandler>	Convert List <media> to List<mediahanlder></mediahanlder></media>
2	checkEmpty	Void	Checks if the filtered media items list is empty and handles displaying a message accordingly
3	updateMediaDisplay	Void	Update and manage the display of media items on the home screen based on the current page

4	showNextMedia	Void	Displays the next set of media items on the screen based on pagination
5	showPreviousMedia	Void	Displays the previous set of media items on the screen based on pagination
6	addMediaHome	Void	Populates the home screen with media items passed as a list.
7	addMenuItems	Void	Adds menu items based on specified text and position to the given menu button

5.3.2. MediaDetailHandler

DetailScreenHandle

- title : String - type : String - category : String - price : int - quantity : int

+ requestToDetail() : void

Attribute

#	Name	Data type	Default value	Description	
1	title	String	NULL	Title of media	
2	type	String	NULL	Type of media	
3	category	String	NULL	Category of media	
4	price	int	0	Price of media	
5	quantity	int	0	Remaining quantity of media	

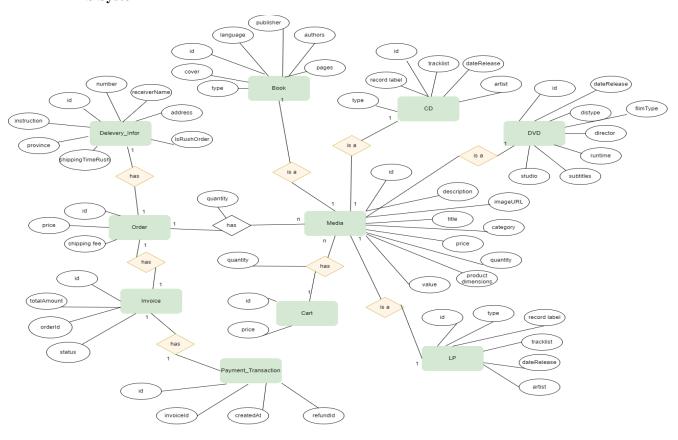
Operation

#	Name	Return type	Description (purpose)	
1	requestToDetail	void	Request to detail media	

6. Data Modeling

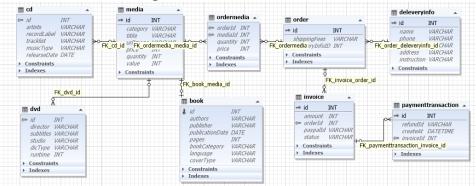
6.1. Conceptual Data Model

AIMS System ERD



6.2. Database Design

5.2.1. Logical Data Model



5.2.2. Physical Data Model

- Media

#	PK	FK	Column Name	Data type	Mandatory	Description
1.	X		id	int	yes	ID, auto increment
2.			title	Varchar(45)	yes	Product's name
3.			category	Varchar(45)	yes	Media type, eg., cd, DVD
4.			value	int	yes	Value of the product
5.			price	int	yes	Current price
6.			quantity	int	yes	Number of products
7.			productDimensions	Varchar(45)	yes	horizontal, length, width dimensions
8.			description	Varchar(45)	yes	Product description
9.			imageURL	Varchar(45)	yes	Product image path
10.			createAt	timestamp	yes	The time the product is added to the system
11.			updateAt	timestamp	no	The time the product is updated to the system

- **CD**

#	PK	FK	Column Name	Data type	Mandatory	Description
1.		X	id	int	yes	ID, same as ID of Media of twhich type is CD
2.			type	Varchar(45)	yes	Music genres
3.			artist	Varchar(45)	yes	Artist's name
4.			dateRelease	datetime	No	Release date
5.			recordLabel	Varchar(45)	yes	Record label

- LD

#	PK	FK	Column Name	Data type	Mandatory	Description
1.		X	id	int	yes	ID, same as ID of Media of twhich type is CD
2.			type	Varchar(45)	yes	Music genres
3.			artist	Varchar(45)	yes	Artist's name
4.			dateRelease	datetime	No	Release date
5.			recordLabel	Varchar(45)	yes	Record label

- Book

#	PK	FK	Column Name	Data type	Mandatory	Description
1.		X	id	int	yes	ID, same as ID of Media of which type is Book
2.			authors	Varchar(45)	yes	Authors of the book
3.			publisher	Varchar(45)	yes	Publishing house
4.			language	Varchar(45)	yes	Language
5.			type	Varchar(45)	yes	Cover type
6.			cover	Varchar(45)	yes	Book cover
7.			page	int	yes	Page number
8.			publishDate	datetime	yes	Date of publishing

- dvd

#	PK	FK	Column Name	Data type	Mandatory	Description
1.		X	id	int	yes	ID, same as ID of Media of which type is DVD
2.			discType	VARCHAR(45)	yes	Disc type
3.			director	VARCHAR(45)	Yes	Director
4.			runtime	int	Yes	Duration
5.			subtitles	VARCHAR(45)	Yes	Subtitles
6.			studio	VARCHAR(45)	yes	Manufacturer
7.			releaseDate	Datetime	Yes	Release date
8.			filmType	VARCHAR(45)	yes	Genres

- deliveryinfo

#	PK	FK	Column Name	Data type	Mandatory	Description
1.	X		id	int	yes	ID, auto increment
2.			receiverName	Varchar(45)	yes	Receiver name
3.			number	Varchar(10)	yes	Receiver phone number
4.			province	Varchar(45)	yes	Provinces
5.			address	Varchar(45)	yes	Delivery address
6.			instruction	Varchar(100)	yes	Delivery instructions
7.			isRushOrder	tinyint(1)	yes	Is Place Rush Order

8.	shippingTimeRush	datetime	no	Delivery Time for RO
9.	createAt	timestamp	yes	
10.	updateAt	timestamp	yes	

- order

#	PK	FK	Column Name	Data type	Mandatory	Description
1.	X		id	int	yes	Id, auto increment
2.			shippingFee	int	yes	Shipping Fee
3.			price	int	yes	Selling price
4.			totalPrice	int	yes	Selling price + VAT
5.		X	deliveryId	int	yes	Delivery Info ID
6.			createAt	timestamp		
7.			updateAt	timestamp		

- order_media

#	PK	FK	Column Name	Data type	Mandatory	Description
1.		X	orderId	int	yes	Order ID
2.		X	mediaId	int	yes	Media ID
3.			quanity	int	yes	Number
4.			price	int	yes	Selling price

- invoice

#	PK	FK	Column Name	Data type	Mandatory	Description
1.	X		id	int	yes	ID
2.			amount	int	yes	Total
3.		X	orderId	int	yes	Order ID
4.			staus	Varchar(45)	yes	Order status

- paymenttransaction

#		PK	FK	Column Name	Data type	Mandatory	Description
5	•	X		id	int	yes	ID
6	•			createAt	timestamp	yes	Date of creation

7.		refundId	Varchar(45)	yes	Transaction contents
8.	X	invoiceId	int	yes	Invoice ID

SQL:

```
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 "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 "Order" (
"id"
      INTEGER NOT NULL,
"name" VARCHAR(45) NOT NULL,
"address"
            VARCHAR(45) NOT NULL,
"phone" VARCHAR(45) NOT NULL,
"shipping fee" INTEGER NOT NULL,
PRIMARY KEY("id" AUTOINCREMENT)
CREATE TABLE IF NOT EXISTS "Transaction" (
"id"
      INTEGER NOT NULL,
"invoiceId"
            INTEGER NOT NULL,
            DATETIME NOT NULL,
"createAt"
"refundId"
            VARCHAR(45) NOT NULL,
CONSTRAINT "fk transaction invoice" FOREIGN KEY("orderId") REFERENCES "Invoice"("id"),
PRIMARY KEY("id" AUTOINCREMENT)
);
CREATE TABLE IF NOT EXISTS "Invoice" (
      INTEGER NOT NULL,
"orderId"
            INTEGER,
"amount"
            INTEGER,
"paypalId"
            VARCHAR(50),
"status" VARCHAR(50),
CONSTRAINT "fk invoice order" FOREIGN KEY("orderId") REFERENCES "Order"("id"),
PRIMARY KEY("id" AUTOINCREMENT)
);
COMMIT;
```