

**ĐẠI HỌC BÁCH KHOA HÀ NỘI**

**TRƯỜNG CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG**

**---**

**o0o**

**--**

**-**

BÁO CÁO THỰC HÀNH

IT3103-744527-2024.1

**BÀI THỰC HÀNH – LAB04**

**Họ và tên: Lê Quang Khải**

**MSSV: 20225638**

**Lớp: VN03-K67**

**GVHD: Lê Thị Hoa**

**HTGD: Đặng Mạnh Cường**

Contents

1. [Create the Book class 4](#_bookmark0)
2. [Creating the abstract Media class 6](#_bookmark3)
3. [Creating the CompactDisc class 8](#_bookmark6)
   1. [Create the Disc class extending the Media clas 8](#_bookmark7)
   2. [Create the Track class which models a track on a compact disc and will store information](#_bookmark11) [incuding the title and length of the track. 10](#_bookmark11)
   3. [Open the CompactDisc class 11](#_bookmark13)
4. [Create the Playable interface 13](#_bookmark16)
5. [Update the Cart class to work with Media 14](#_bookmark21)
6. [Update the Store class to work with Media 18](#_bookmark26)
7. [Constructors of whole classes and parent classes 20](#_bookmark29)
8. [Unique item in a list 21](#_bookmark33)
9. [Polymorphism with toString() method 22](#_bookmark37)
10. [Sort media in the car 23](#_bookmark41)
11. [Create a complete console application in the Aims class 24](#_bookmark45)
    1. [Người dùng chọn 1: View store 25](#_bookmark47)
       1. [Người dùng tiếp tục chọn 1. See a media’s details 26](#_bookmark49)
       2. [Người dùng chọn 2: Add a media to the cart 27](#_bookmark52)
       3. [Người dùng chọn 3: Play a media 27](#_bookmark54)
       4. [Người dùng chọn 4: See current cart 28](#_bookmark56)
    2. [Người dùng chọn 2: Update store 29](#_bookmark58)
       1. [Người dùng chọn 1: Add a media to the store 29](#_bookmark60)
       2. [Người dùng chọn 2: Remove a media from the store 30](#_bookmark63)
    3. [Người dùng chọn 3: See current cart 31](#_bookmark66)
       1. [Người dùng chọn 1: Filter medias in cart 32](#_bookmark69)
       2. [Người dùng chọn 2: Sort medias in cart 34](#_bookmark72)
       3. [Người dùng chọn 3: Remove media from cart 35](#_bookmark75)
       4. [Người dùng chọn 4: Play a media 36](#_bookmark78)
       5. [Người dùng chọn 5: Place order 36](#_bookmark80)
12. [Class Diagram 38](#_bookmark83)
13. [UseCase Diagram 39](#_bookmark85)
14. [Answer Questions 39](#_bookmark86)

Table of Figures

[Figure 1.1: Book Class 1 4](#_bookmark1)

[Figure 1.2: Book Class 2 5](#_bookmark2)

[Figure 2.1: Media Class 1 6](#_bookmark4)

[Figure 2.2: Media Class 2 7](#_bookmark5)

[Figure 3.1: Disc Class 8](#_bookmark8)

[Figure 3.2: DigitalVideoDisc Class 9](#_bookmark9)

[Figure 3.3: CompactDisc Class 9](#_bookmark10)

[Figure 3.4: Track Class 10](#_bookmark12)

[Figure 3.5: CompactDisc Class 1 11](#_bookmark14)

[Figure 3.6: CompactDisc Class 2 12](#_bookmark15)

[Figure 4.1: Playable interface 13](#_bookmark17)

[Figure 4.2: Method play() của DigitalVideoDisc 13](#_bookmark18)

[Figure 4.3: Method play() của Track 13](#_bookmark19)

[Figure 4.4: Method play() của CompactDisc 13](#_bookmark20)

[Figure 5.1: Cart Class 1 14](#_bookmark22)

[Figure 5.2: Cart Class 2 15](#_bookmark23)

[Figure 5.3: Cart Class 3 16](#_bookmark24)

[Figure 5.4: Cart Class 4 17](#_bookmark25)

[Figure 6.1: Store Class 1 18](#_bookmark27)

[Figure 6.2: Store Class 2 19](#_bookmark28)

[Figure 7.1: Constructor Track Class 20](#_bookmark30)

[Figure 7.2: Constructor CompactDisc Class 20](#_bookmark31)

[Figure 7.3: Constructor Media Class 20](#_bookmark32)

[Figure 7.4: Constructor Disc Class 21](#_bookmark34)

[Figure 8.1: Override equals in Media Class 21](#_bookmark35)

[Figure 8.2: Override equals in Track Class 21](#_bookmark36)

[Figure 9.1: Code mô phỏng Polymorphism 22](#_bookmark38)

[Figure 9.2: Override toString() in Media Class 22](#_bookmark39)

[Figure 9.3: Result demo Polymorphism 22](#_bookmark40)

[Figure 10.1: Add the comparators as attributes of the Media class 23](#_bookmark42)

[Figure 10.2: MediaComparatorByCostTitle Class 23](#_bookmark43)

[Figure 10.3: MediaComparatorByTitleCost Class 24](#_bookmark44)

[Figure 11.1: Màn hình chính 24](#_bookmark46)

[Figure 11.2: Vào Trang View Store 25](#_bookmark48)

[Figure 11.3: See a media's details 26](#_bookmark50)

[Figure 11.4: Thêm vào Cart 26](#_bookmark51)

[Figure 11.5: Thêm media vào Cart 27](#_bookmark53)

[Figure 11.6: Play a media 27](#_bookmark55)

[Figure 11.7: See current cart after sort 28](#_bookmark57)

[Figure 11.8: Vào Trang Update Store 29](#_bookmark59)

[Figure 11.9: Add a media to store 29](#_bookmark61)

[Figure 11.10: Result after add media to store 30](#_bookmark62)

[Figure 11.11: Remove a media from the store 30](#_bookmark64)

[Figure 11.12: Result after remove a media 31](#_bookmark65)

[Figure 11.13: Vào trang See current cart 31](#_bookmark67)

[Figure 11.14: Media in Cart 32](#_bookmark68)

[Figure 11.15: Filter Cart By id 32](#_bookmark70)

[Figure 11.16: Filter Cart By Title 33](#_bookmark71)

[Figure 11.17: Sort Cart By Title 34](#_bookmark73)

[Figure 11.18: Sort Cart By Cost 34](#_bookmark74)

[Figure 11.19: Remove media by id 35](#_bookmark76)

[Figure 11.20: Result after remove media in cart by id 35](#_bookmark77)

[Figure 11.21: Play a media in cart 36](#_bookmark79)

[Figure 11.22: Order 36](#_bookmark81)

[Figure 11.23: Result after order 37](#_bookmark82)

[Figure 12.1: Class Diagram 38](#_bookmark84)

[Figure 13.1: UseCase Diagram 39](#_bookmark87)

[Figure 14.1: Triển khai Comparable trong lớp trừu tượng Media 40](#_bookmark88)

[Figure 14.2: Mở rộng để so sánh nhiều thuộc tính hơn 40](#_bookmark89)

[Figure 14.3: Triển khai tại lớp con 40](#_bookmark90)

# Create the Book class

# A screenshot of a computer screen Description automatically generated

*Figure 1.1: Book Class 1*

A screenshot of a computer program

Description automatically generated

*Figure 1.2: Book Class 2*

# Creating the abstract Media class

Đây sẽ là lớp cha để các lớp DigitalVideoDisc, Book kế thừa.

A screenshot of a computer program

Description automatically generated

*Figure 2.1: Media Class 1*

A screenshot of a computer program

Description automatically generated

*Figure 2.2: Media Class 2*

# Creating the CompactDisc class

## A screenshot of a computer program Description automatically generatedCreate the Disc class extending the Media clas

*Figure 3.1: Disc Class*

A screenshot of a computer program

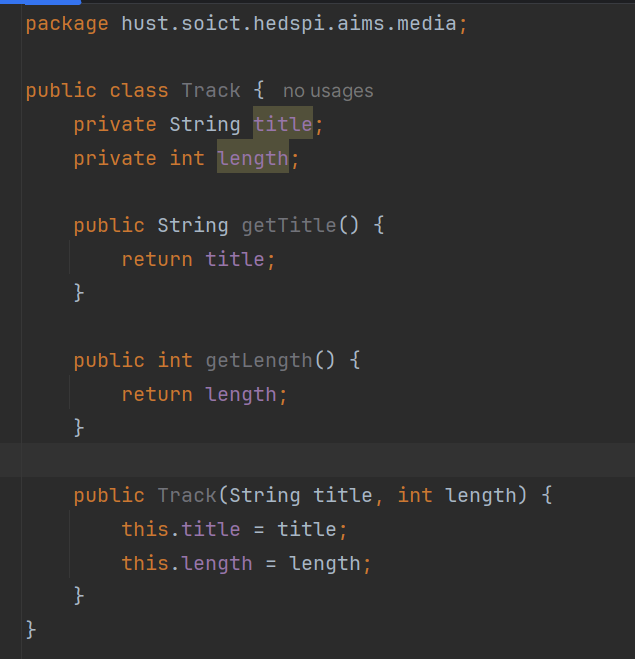
Description automatically generated

*A screenshot of a computer program

Description automatically generatedFigure 3.2: DigitalVideoDisc Class*

*Figure 3.3: CompactDisc Class*

## Create the Track class which models a track on a compact disc and will store information incuding the title and length of the track.



*Figure 3.4: Track Class*

## A screen shot of a computer Description automatically generatedOpen the CompactDisc class

*Figure 3.5: CompactDisc Class 1*

A screen shot of a computer program

Description automatically generated

*Figure 3.6: CompactDisc Class 2*

# A screen shot of a computer Description automatically generatedCreate the Playable interface

*Figure 4.1: Playable interface*

Implement play() cho các class DigitalVideoDisc, Track, CompactDisc

A screen shot of a computer program

Description automatically generated

*Figure 4.2: Method play() của DigitalVideoDisc*

*A screen shot of a computer program

Description automatically generated*

*Figure 4.3: Method play() của Track*

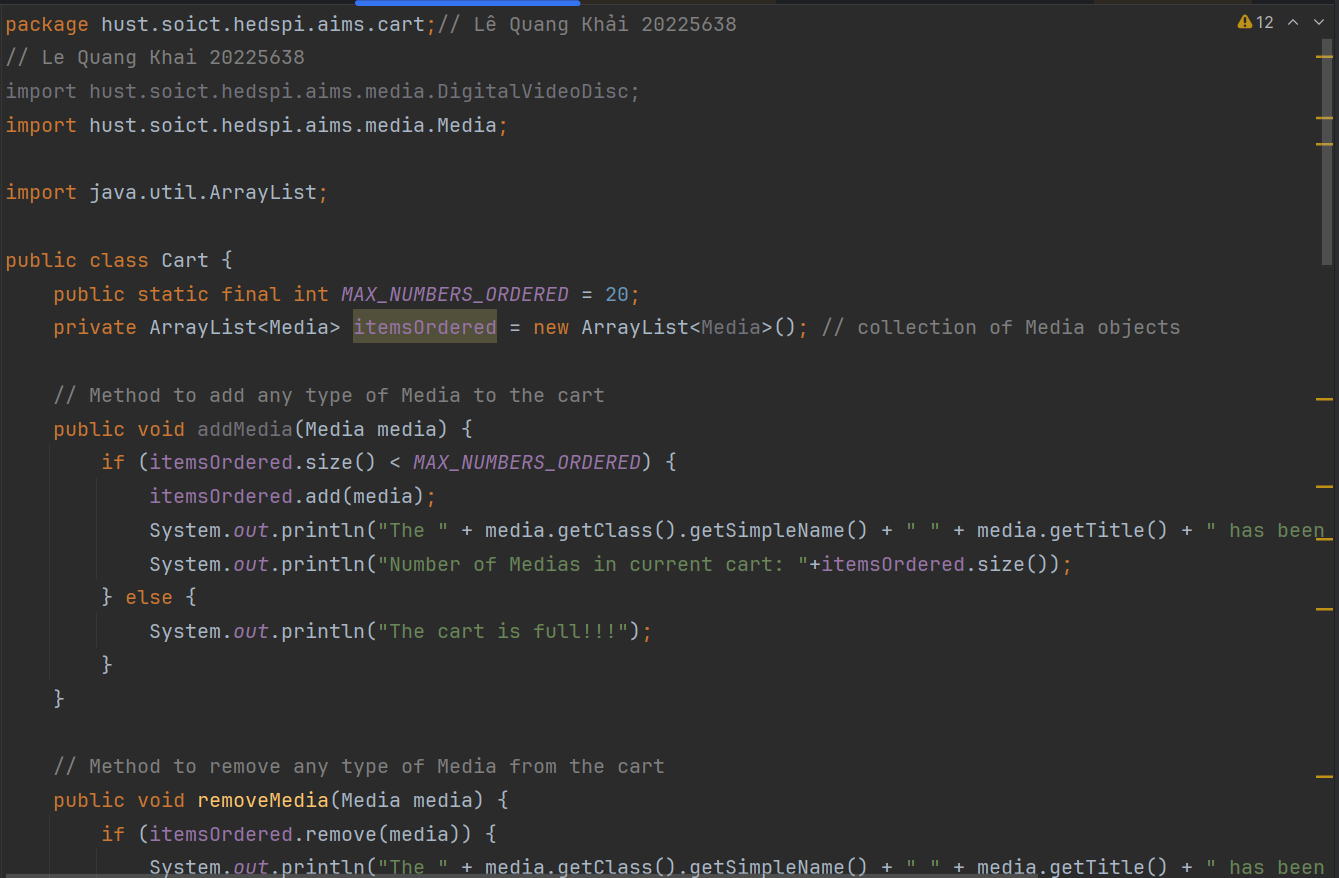
*A computer screen shot of a computer code

Description automatically generated*

*Figure 4.4: Method play() của CompactDisc*

# Update the Cart class to work with Media

Lớp Cart bây giờ cần có khả năng tương tác với các đối tượng DVD, CD và Book. Vì các lớp DVD, CD và Book đều kế thừa từ lớp Media, nên thay vì làm việc trực tiếp với từng lớp con, lớp cart chỉ cần giao tiếp với lớp Media là có thể hoạt động được với tất cả.



*Figure 5.1: Cart Class 1*

A screen shot of a computer program

Description automatically generated

*Figure 5.2: Cart Class 2*

A computer screen shot of a program

Description automatically generated

*Figure 5.3: Cart Class 3*

A screenshot of a computer program

Description automatically generated

*Figure 5.4: Cart Class 4*

# Update the Store class to work with Media

# A screen shot of a computer program Description automatically generated

*Figure 6.1: Store Class 1*

A screen shot of a computer

Description automatically generated

*Figure 6.2: Store Class 2*

# A black background with white text Description automatically generatedConstructors of whole classes and parent classes

*Figure 7.1: Constructor Track Class*

*A computer screen shot of a black background

Description automatically generated*

*Figure 7.2: Constructor CompactDisc Class*

Lớp Disc kế thừa lớp Media, khi đó lớp Media là lớp cha, lớp Disc là lớp con.

A computer screen shot of a program code

Description automatically generated

*Figure 7.3: Constructor Media Class*

A computer screen shot of a program

Description automatically generated

# Unique item in a list

*Figure 7.4: Constructor Disc Class*

Để tránh trùng lặp các phần tử media trong giỏ hàng hoặc các track trong một đĩa CD, chúng ta có thể ghi đè lại phương thức equals() mặc định kế thừa từ lớp Object. Việc này cho phép so sánh bản chất thay vì so sánh vị trí ô nhớ của các đối tượng, qua đó ngăn chặn thêm các phần tử bị trùng lắp vào danh sách.

A computer screen with text

Description automatically generated

*Figure 8.1: Override equals in Media Class*

*A screen shot of a computer program

Description automatically generated*

*Figure 8.2: Override equals in Track Class*

# Polymorphism with toString() method

# A screen shot of a computer Description automatically generated

*Figure 9.1: Code mô phỏng Polymorphism*

*A screen shot of a computer code

Description automatically generated*

*Figure 9.2: Override toString() in Media Class*

Kết quả

A screenshot of a computer program

Description automatically generated

*Figure 9.3: Result demo Polymorphism*

Lớp Media là lớp cơ sở được kế thừa bởi các lớp cụ thể hơn là CompactDisc, DigitalVideoDisc và Book. Khi khởi tạo các đối tượng cd, dvd, book thuộc lớp con rồi gán chúng cho biến kiểu Media, ta áp dụng kỹ thuật gọi là upcasting.

Việc thêm chúng vào danh sách media và duyệt danh sách để in ra thông tin mỗi phần tử bằng phương thức toString() là ví dụ điển hình cho tính đa hình động. Mỗi lớp con có thể cài đặt riêng toString() nên kết quả sẽ khác nhau dựa theo loại đối tượng, mà không cần quan tâm đến kiểu cụ thể của từng phần tử.

# Sort media in the car

Sắp xếp các media trong giỏ hàng theo hai tiêu chí:

* Bằng title: Hiển thị tất cả các media theo thứ tự bảng chữ cái. Trong trường hợp cùng title,

media có cost cao hơn sẽ được hiển thị trước.

* Bằng cost: Hiển thị theo thứ tự cost giảm dần. Trong trường hợp cost như nhau, sắp xếp media theo thứ tự bảng chữ cái



*Figure 10.1: Add the comparators as attributes of the Media class*

*A computer screen shot of a program

Description automatically generated*

*Figure 10.2: MediaComparatorByCostTitle Class*

A screen shot of a computer program

Description automatically generated

*Figure 10.3: MediaComparatorByTitleCost Class*

# Create a complete console application in the Aims class

*Figure 11.1: Màn hình chính*

## Người dùng chọn 1: View store

## **A screenshot of a computer program Description automatically generated**

*Figure 11.2: Vào Trang View Store*

### A screenshot of a computer program Description automatically generatedNgười dùng tiếp tục chọn 1. See a media’s details

*Figure 11.3: See a media's details*

*A screenshot of a computer

Description automatically generated*

*Figure 11.4: Thêm vào Cart*

### A screenshot of a computer program Description automatically generatedNgười dùng chọn 2: Add a media to the cart

*Figure 11.5: Thêm media vào Cart*

### A screenshot of a computer program Description automatically generatedNgười dùng chọn 3: Play a media

*Figure 11.6: Play a media*

### A screenshot of a computer program Description automatically generatedNgười dùng chọn 4: See current cart

*Figure 11.7: See current cart*

## A screenshot of a computer program Description automatically generatedNgười dùng chọn 2: Update store

*Figure 11.8: Vào Trang Update Store*

### A screenshot of a computer program Description automatically generatedNgười dùng chọn 1: Add a media to the store

*Figure 11.9: Add a media to store*

A screen shot of a computer

Description automatically generated=> Kết quả sau khi thêm

*Figure 11.10: Result after add media to store*

### A screenshot of a computer Description automatically generatedNgười dùng chọn 2: Remove a media from the store

*Figure 11.11: Remove a media from the store*

A screen shot of a computer

Description automatically generated=> Kết quả sau khi remove

*Figure 11.12: Result after remove a media*

## Người dùng chọn 3: See current cart

*Figure 11.13: Vào trang See current cart*

### A screenshot of a computer program Description automatically generatedNgười dùng chọn 1: Filter medias in cart

*Figure 11.15: Filter Cart By id*

A screen shot of a computer

Description automatically generated*Figure 11.16: Filter Cart By Title*

### A screenshot of a computer Description automatically generatedNgười dùng chọn 2: Sort medias in cart

*Figure 11.17: Sort Cart By Title*

*A screenshot of a computer

Description automatically generated*

*Figure 11.18: Sort Cart By Cost*

### A screenshot of a computer program Description automatically generatedNgười dùng chọn 3: Remove media from cart

*Figure 11.19: Remove media by title*

A screenshot of a computer

Description automatically generated=> Kết quả

*Figure 11.20: Result after remove media in cart by id*

### A screenshot of a computer program Description automatically generatedNgười dùng chọn 4: Play a media

*Figure 11.21: Play a media in cart*

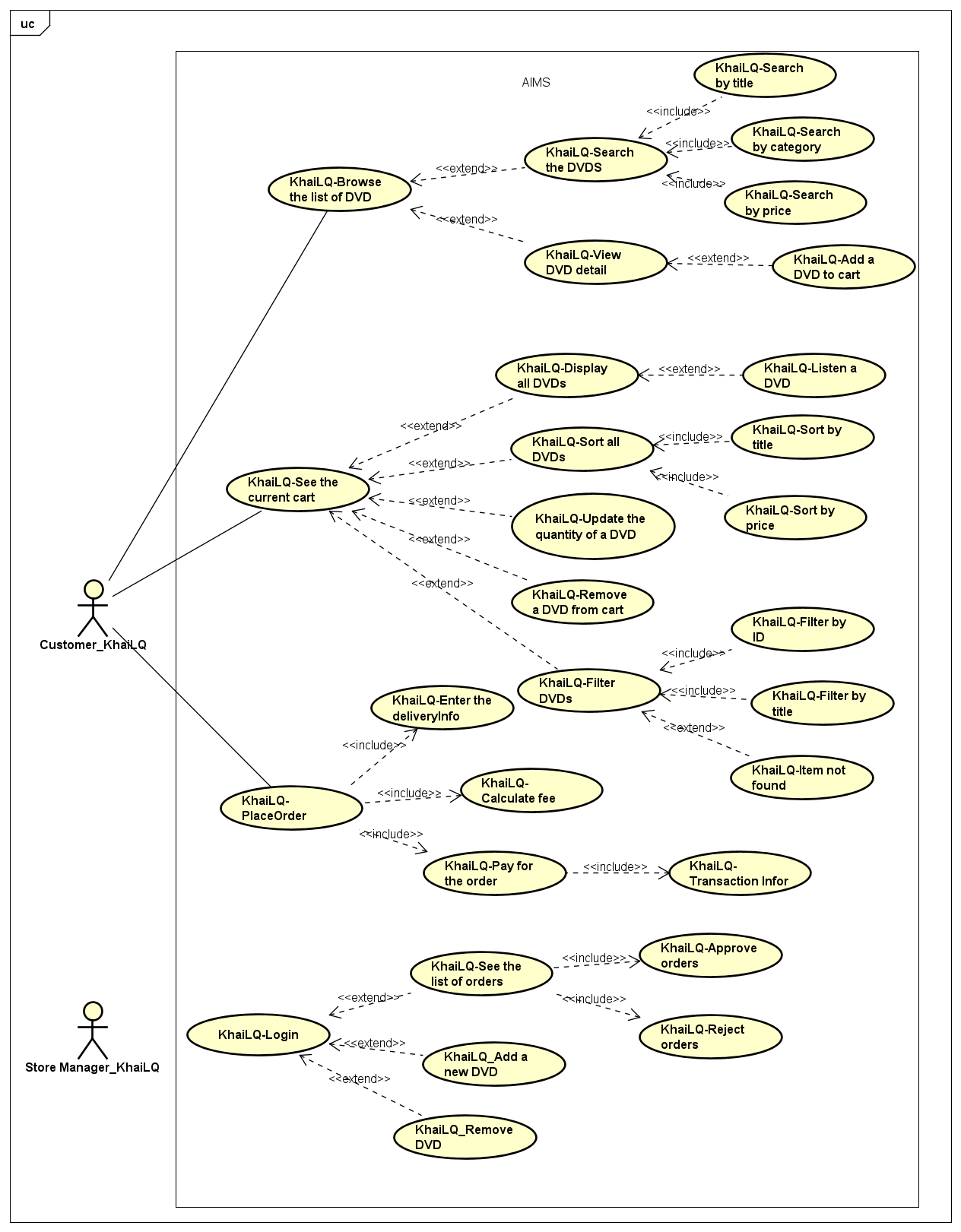
### A screenshot of a computer Description automatically generatedNgười dùng chọn 5: Place order

*Figure 11.22: Order*

# Class Diagram

*Figure 12.1: Class Diagram*

1. UseCase Diagram



1. Answer Questions

*Figure 13.1: UseCase Diagram*

### **10. Unique Item in a List**

### Nếu đối tượng được truyền vào không phải là instance của lớp Media, việc so sánh sẽ không thể thực hiện được. Tuy nhiên, nếu đối tượng đó là instance của một lớp con kế thừa từ Media, phép so sánh vẫn hoạt động như bình thường.

### Ví dụ minh họa trong lớp Aims:

### Media book1 = new Book(5, "Book1", "Horror", 4.5f);

### Media book2 = new Book(6, "Book1", "Scientific", 5.5f);

### if (book1.equals(book2)) {

### System.out.println("Two objects are equal");

### } else {

### System.out.println("Two objects are not equal");

### }

### **Kết quả**: Two objects are equal, vì cả hai đối tượng đều có tiêu đề (title) giống nhau, và phép so sánh chỉ dựa trên thuộc tính này.

### **1. Lớp nào nên triển khai giao diện Comparable?**

Lớp đại diện cho các mặt hàng (ví dụ: Item, Media, hoặc lớp con cụ thể như Book, DVD) nên triển khai giao diện Comparable. Điều này phụ thuộc vào cách cấu trúc lớp của chương trình, nhưng thường lớp cơ bản (ví dụ: Media) sẽ là lớp thích hợp để triển khai.

### **2. Làm thế nào để triển khai phương thức compareTo() để phản ánh thứ tự mong muốn?**

Trong lớp triển khai, bạn cần override phương thức compareTo() để xác định logic so sánh.  
Ví dụ, nếu muốn sắp xếp theo **title** trước rồi đến **cost**, có thể viết như sau:

@Override

public int compareTo(Media other) {

int titleComparison = this.title.compareTo(other.title);

if (titleComparison != 0) {

return titleComparison; // Sắp xếp theo title

}

return Double.compare(this.cost, other.cost); // Nếu title giống nhau, sắp xếp theo cost

}

### **3. Có thể có hai quy tắc sắp xếp (theo title rồi cost và theo cost rồi title) nếu dùng Comparable không?**

Không, giao diện Comparable chỉ cho phép định nghĩa **một quy tắc sắp xếp duy nhất** thông qua phương thức compareTo().  
Nếu cần nhiều quy tắc sắp xếp, bạn nên sử dụng Comparator thay vì Comparable.

### **4. Nếu DVD có quy tắc sắp xếp khác (theo title, rồi đến độ dài giảm dần, rồi đến cost), làm thế nào để sửa đổi mã?**

Để xử lý quy tắc sắp xếp riêng cho DVD, bạn cần override phương thức compareTo() trong lớp DVD và triển khai logic so sánh riêng:

@Override

public int compareTo(Media other) {

if (other instanceof DVD) {

DVD otherDVD = (DVD) other;

int titleComparison = this.title.compareTo(otherDVD.title);

if (titleComparison != 0) {

return titleComparison; // Sắp xếp theo title

}

int lengthComparison = Integer.compare(otherDVD.length, this.length);

if (lengthComparison != 0) {

return lengthComparison; // Sắp xếp theo độ dài giảm dần

}

return Double.compare(this.cost, otherDVD.cost); // Nếu cả title và length giống nhau, sắp xếp theo cost

}

// Nếu không phải DVD, xử lý theo logic mặc định

return super.compareTo(other);

}