

Báo Cáo Thực Hành Lập Trình Hướng Đối Tượng Lab-03

Phan Thanh Thắng - 20225927

Ngày 24 tháng 11 năm 2024

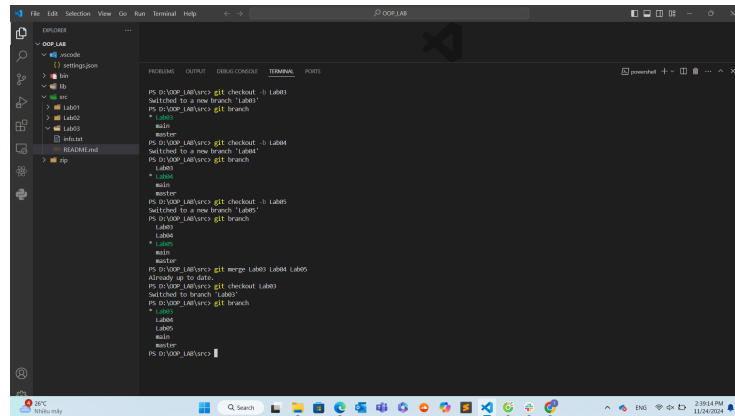
Mục lục

1	Tạo các branch trong project	3
1.1	Tạo branch con Lab03, Lab04, Lab05 trong branch cha trong main	3
1.2	Tạo 9 branch trong Lab03 theo yêu cầu	3
1.3	Trộn branch	4
1.4	Push lên github	4
2	Working with method overloading	5
2.1	Overloading by differing types of parameter	5
2.2	Overloading by differing the number of parameters	6
3	Passing parameter	6
3.1	Trả lời 2 câu hỏi (Tiếng Việt)	6
3.2	Tạo class TestPassingParameter và kết quả khi chạy	7
3.3	Giải quyết vấn đề trong đoạn code này	7
3.3.1	Trả lời 2 câu hỏi (Tiếng Việt)	7
3.3.2	Sửa lại hàm swap	8
4	Sử dụng Debug	8
5	Classifier Member and Instance Member	9
6	Open the Cart class	9
6.1	Sửa code (Thêm hàm print trong cart.java)	9
6.2	Chạy thử và xem kết quả	10
7	Implement the Store class	10
7.1	Tạo lớp Store	10
7.2	Tạo lớp TestStore và kiểm thử	11
8	Re-organize your projects	11

9	String, StringBuilder and StringBuffer	12
9.1	Tạo lớp ConcatenationInLoops	12
9.2	Tạo lớp GarbageCreator	12
9.3	Tạo lớp NoGarbage	13
10	Release flow demonstration	13
11	Cập nhật các Diagram trong project AIMS	14
11.1	Class diagram	14
11.2	Use-case diagram	15

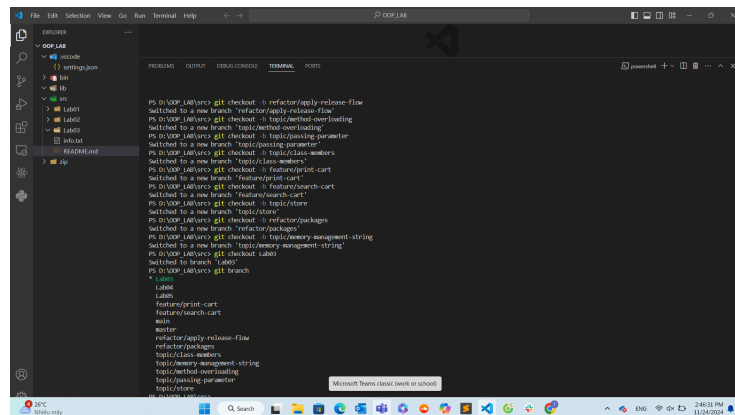
1 Tạo các branch trong project

1.1 Tạo branch con Lab03, Lab04, Lab05 trong branch cha trong main



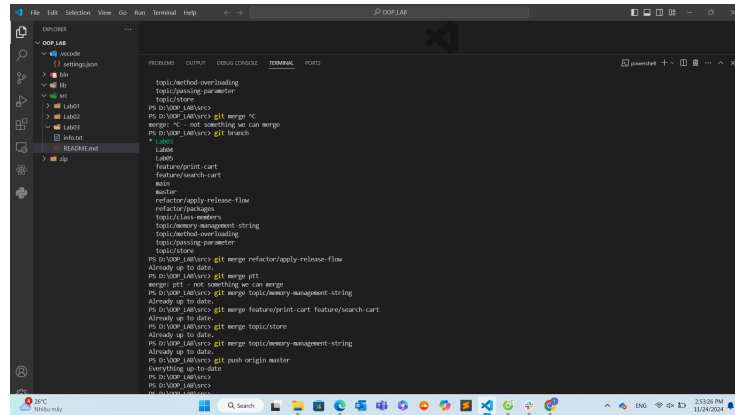
Hình 1: Tạo 3 branch Lab03 Lab04 Lab05 trong main

1.2 Tạo 9 branch trong Lab03 theo yêu cầu



Hình 2: Tạo 9 branch trong Lab03 theo yêu cầu

1.3 Trộn branch



Hình 3: Trộn branch

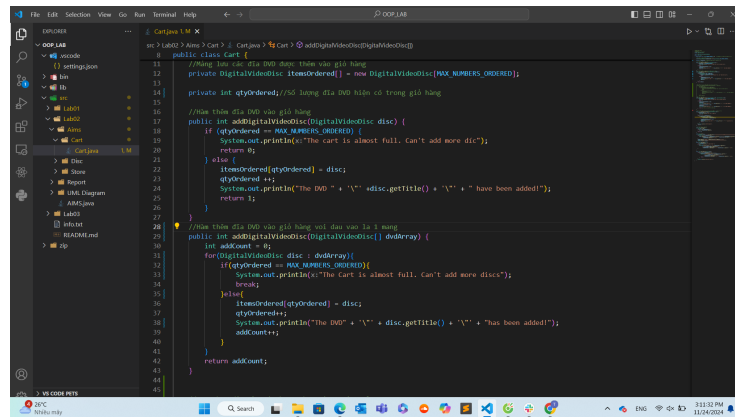
1.4 Push lên github

Cách push: Mở Git Gui, checkout all branch, sau đó chọn các branch rồi up

2 Working with method overloading

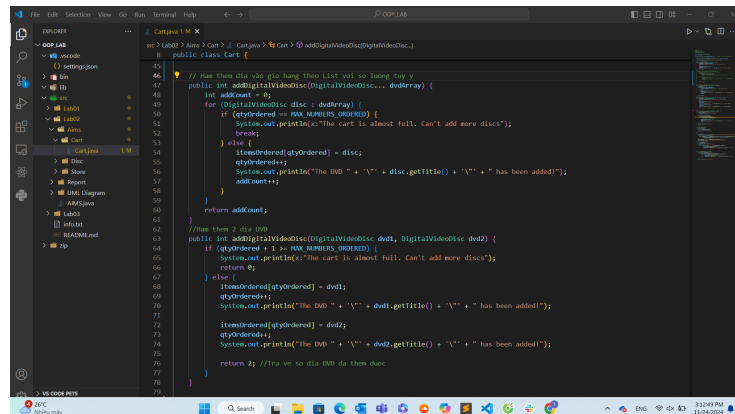
2.1 Overloading by differing types of parameter

Phương thức mới đầu tiên `addDigitalVideoDisc(DigitalVideoDisc[] dvdList)` cho phép thêm một danh sách các đĩa DVD vào giỏ hàng. Phương thức mới thứ hai `addDigitalVideoDisc(DigitalVideoDisc...`



Hình 4:

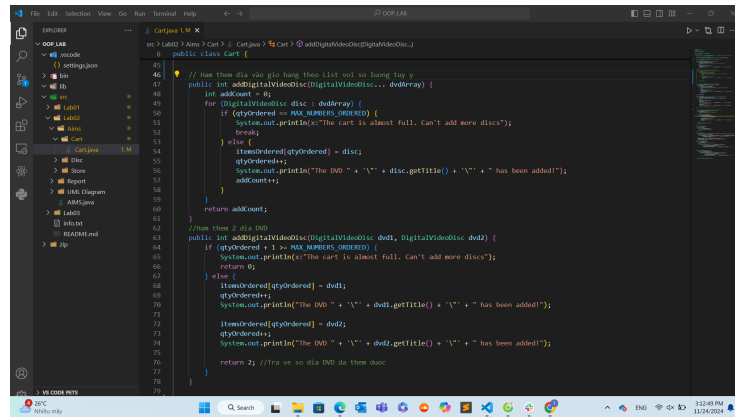
`dvdArray`) sử dụng `varargs` để cho phép thêm một số lượng tùy ý các đĩa DVD vào giỏ hàng. Cách sử dụng `varargs` linh hoạt hơn vì nó



Hình 5:

cho phép thêm bất kỳ số lượng đĩa DVD nào mà không cần chỉ định mảng một cách rõ ràng. Điều này làm cho mã nguồn sạch sẽ hơn và thuận tiện hơn khi gọi phương thức. Vì thế em thích phương thức dùng `varargs`.

2.2 Overloading by differing the number of parameters



Hình 6:

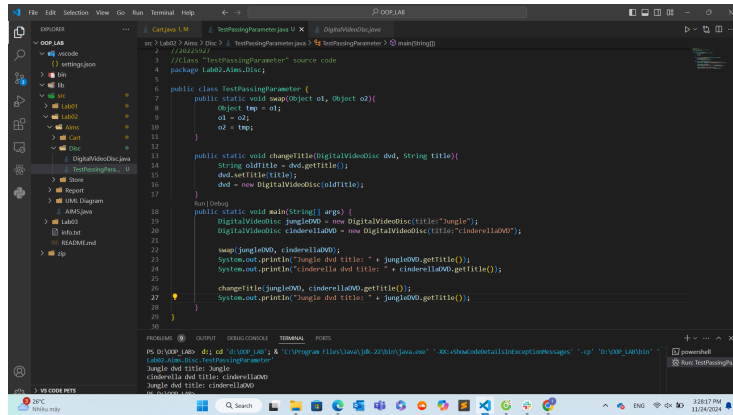
3 Passing parameter

3.1 Trả lời 2 câu hỏi (Tiếng Việt)

Câu hỏi: Java là ngôn ngữ lập trình "Pass by Value" hay "Pass by Reference"?

Trả lời: Java là ngôn ngữ lập trình "Pass by Value". Trong Java, khi bạn truyền một tham số cho một phương thức, giá trị của tham số được sao chép và truyền vào phương thức. Điều này có nghĩa là nếu bạn thay đổi giá trị của tham số bên trong phương thức, giá trị của biến gọi phương thức không bị ảnh hưởng.

3.2 Tạo class TestPassingParameter và kết quả khi chạy



Hình 7:

Có vấn đề trong hàm swap

3.3 Giải quyết vấn đề trong đoạn code này

3.3.1 Trả lời 2 câu hỏi (Tiếng Việt)

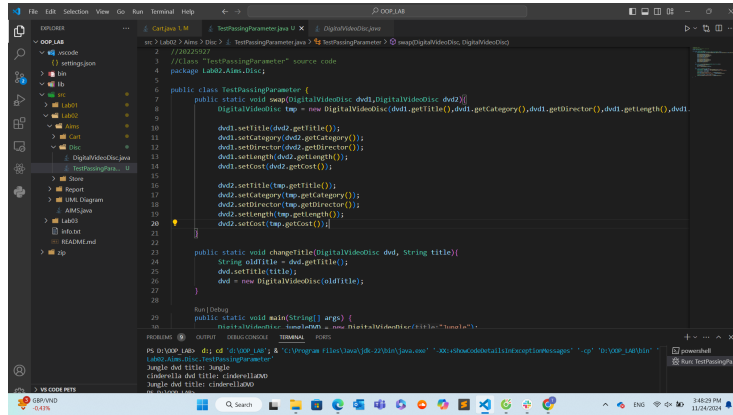
Q: After the call of swap(jungleDVD, cinderellaDVD) why does the title of these two objects still remain?

A: Sau khi thực hiện phương thức swap(jungleDVD, cinderellaDVD), tiêu đề của hai đối tượng vẫn giữ nguyên vì trong Java, tham số được truyền vào phương thức là giá trị của đối tượng, không phải là tham chiếu đến đối tượng. Khi ta thay đổi giá trị của tham số bên trong phương thức (như việc đổi chỗ giữa o1 và o2), sự thay đổi này không ảnh hưởng đến giá trị của các đối tượng gốc.

Q: After the call of changeTitle(jungleDVD, cinderellaDVD.getTitle()) why is the title of the JungleDVD changed?

A: Sau khi gọi changeTitle(jungleDVD, cinderellaDVD.getTitle()), tiêu đề của jungleDVD bị thay đổi vì trong phương thức changeTitle, ta thực hiện thay đổi trực tiếp trên đối tượng dvd (được truyền vào phương thức) bằng cách gọi dvd.setTitle(title). Điều này ảnh hưởng trực tiếp đến đối tượng gốc được truyền vào phương thức.

3.3.2 Sửa lại hàm swap

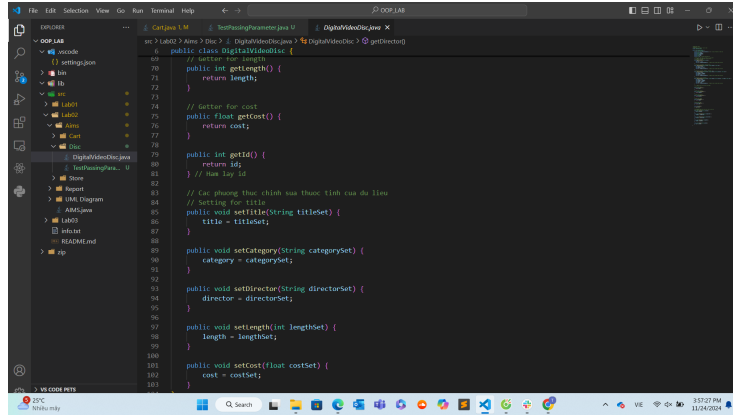


Hình 8:

4 Sử dụng Debug

Em sử dụng Vscode để lập trình Java, trong IDE này nó tích hợp sẵn công cụ Debug tự động nên có hơi khác với Eclipse, và em đã biết sử dụng công cụ Debug thủ công trong IDE này rồi. Nhấn F5 hoặc chọn Run and Debug > Start Debugging. Chương trình sẽ chạy và dừng ở các breakpoints đã đặt.

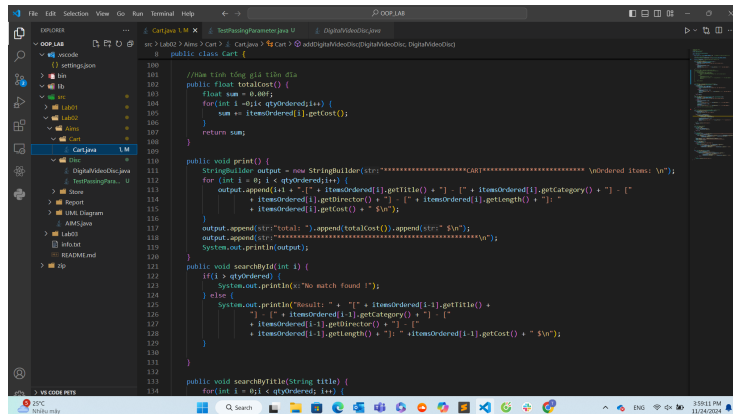
5 Classifier Member and Instance Member



Hình 9:

6 Open the Cart class

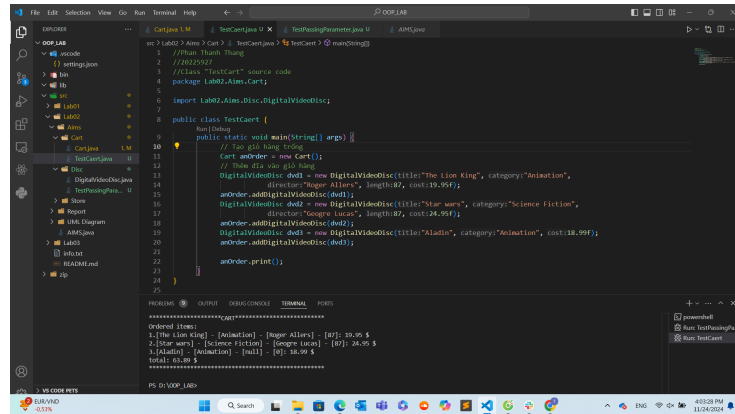
6.1 Sửa code (Thêm hàm print trong cart.java)



Hình 10:

6.2 Chạy thử và xem kết quả

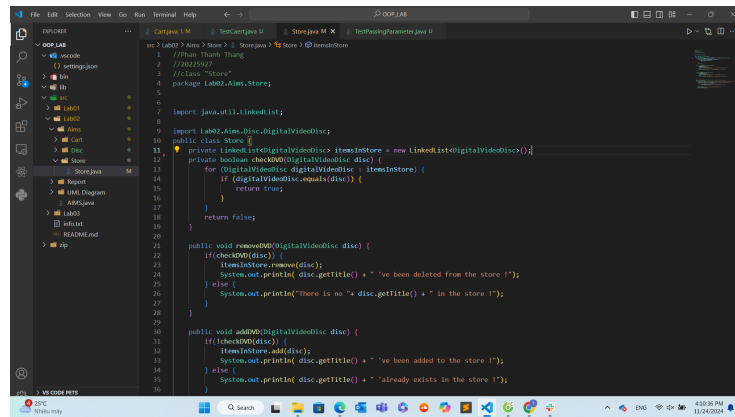
Phương thức `print()` được gọi ở lớp `TestCart`



Hình 11:

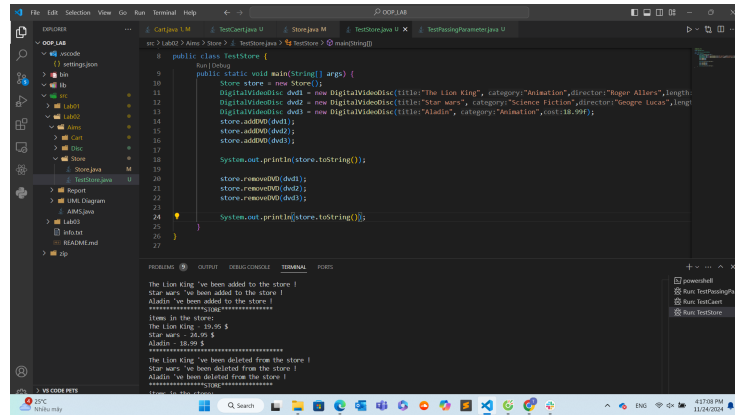
7 Implement the Store class

7.1 Tạo lớp Store



Hình 12:

7.2 Tạo lớp TestStore và kiểm thử



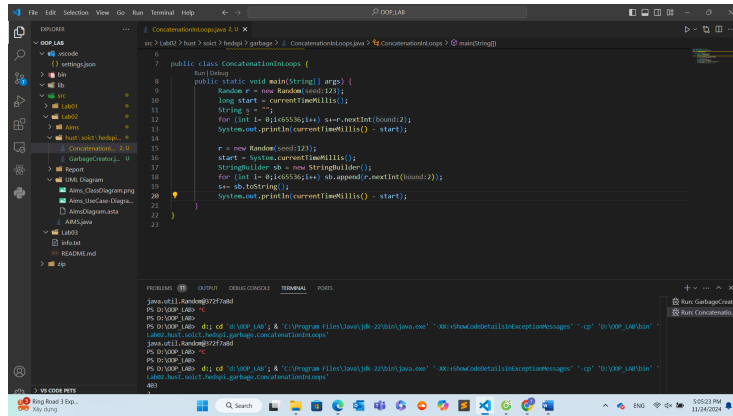
Hình 13:

8 Re-organize your projects

Trong code và trong project em đã làm rồi.

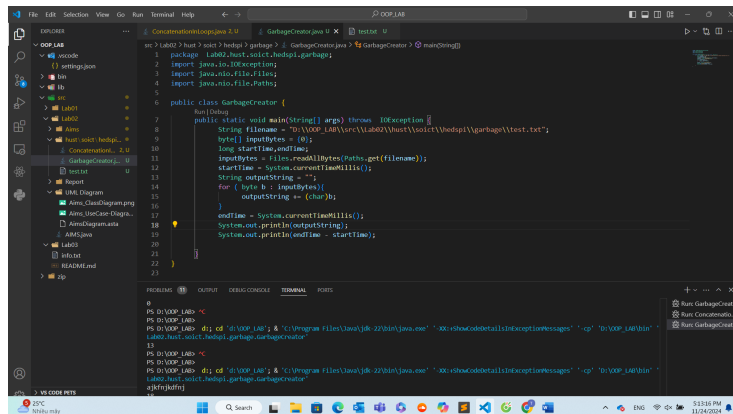
9 String, StringBuilder and StringBuffer

9.1 Tạo lớp ConcatenationInLoops



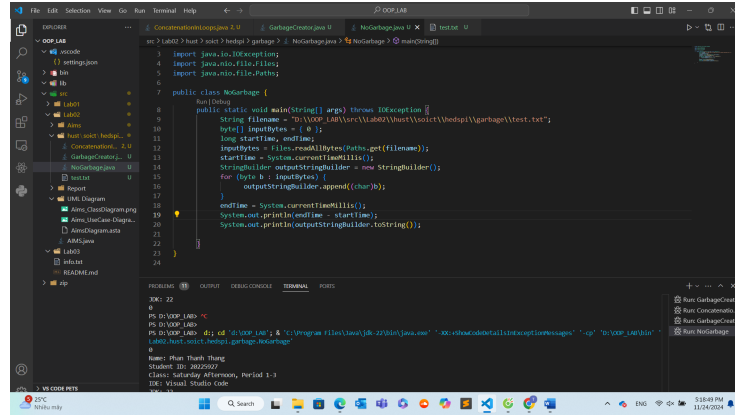
Hình 14:

9.2 Tạo lớp GarbageCreator



Hình 15:

9.3 Tạo lớp NoGarbage



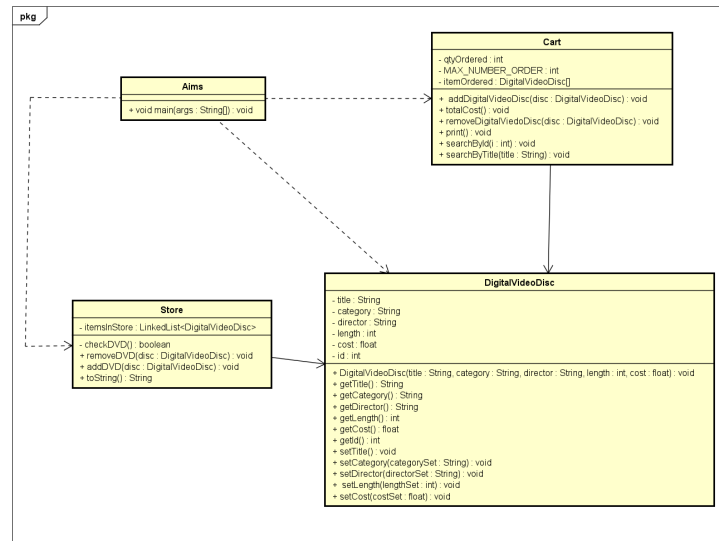
Hình 16:

10 Release flow demonstration

Em đã thử làm cái này rồi

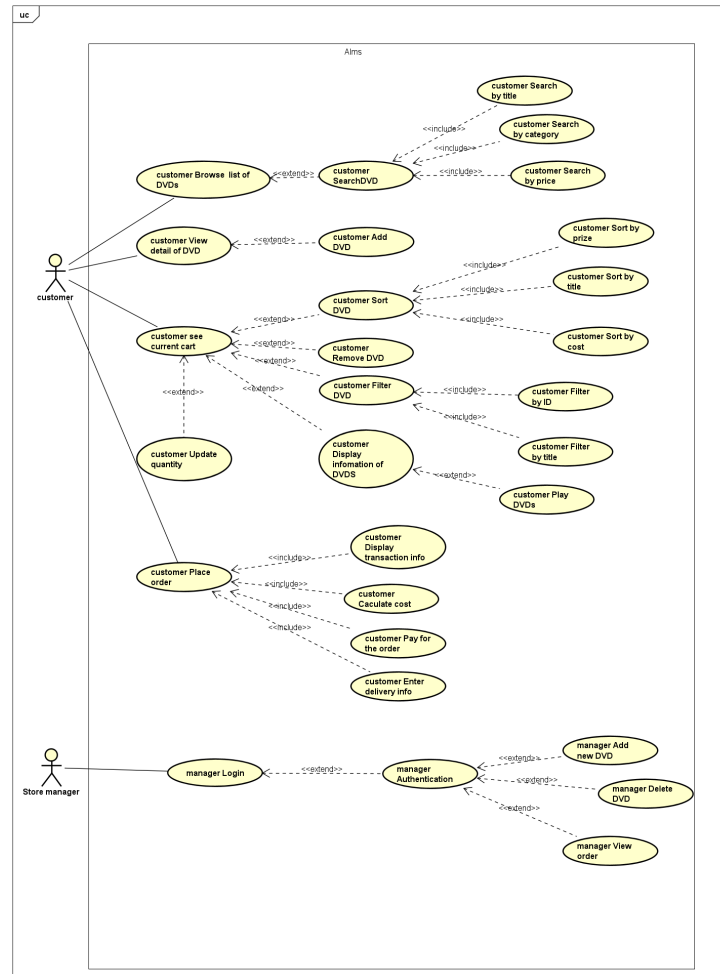
11 Cập nhật các Diagram trong project AIMS

11.1 Class diagram



Hình 17:

11.2 Use-case diagram



Hình 18: