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

BÁO CÁO THỰC HÀNH
IT3103-744527-2024.1

BÀI THỰC HÀNH -LAB02

Họ và tên sv: Chu Đình Hà

MSSV: 20225712

Lớp: Việt Nhật 03

GVHD: Lê Thị Hoa

HTGD: Đặng Mạnh Cường

Hà Nội 10/2024

BÁO CÁO THỰC HÀNH LAB 2

LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG

Table of Contents

Table of Figures

1. Bài toán đặt ra

There might be a future that Tiki and Sendo be in talks over a potential merger to contend other e-commerce platforms and especially those who have foreign backers. The merger of these two firms would create a Ti-do company, where "Ti" is from Tiki, and "do" is from Sendo, which means a billion-dollar company in Vietnamese. That firm, Ti-do company, would like you to help them create a brand-new system for AIMS project (AIMS stands for An Internet Media Store). Currently, there is only one type of media: Digital Video Disc (DVD).

2. Yêu cầu hệ thống

2.1. Đối với Customer:

Customers can browse the list of DVDs available in the store, the display order is based on their added date, from latest to oldest. When a customer wants to search for DVDs to add to cart, he or she can choose one of three searching options. The software will display a list of all matches (latest DVDs first) with all their information. He or she can also choose to play a specific DVD. The software will play a DVD (a demo part). If a DVD has the length 0 or less, the system must notify the customer that the DVD cannot be played.

- When a customer searches for DVDs by title, he or she provides a string of keywords. If any DVD has the title containing any word in the string of keywords, it is counted as a match. Note that the comparison of words here is case-insensitive.
- When a customer searches for DVDs by category, he or she provides the category name. If any DVD has the matching category (case-insensitive), it is counted as a match.
- When a customer searches for DVDs by price, he or she provides either the minimum and maximum cost, or just the maximum cost.

Customers can view the detail information of a DVD from the list of DVDs. He/she can add a DVD to a cart from a list of DVDs or the detail screen.

When a customer wants to see the current cart, the system displays all the information of the DVDs, along with the total cost. Customers may listen to a DVD (a demo part) in the cart before confirming to place an order. Customers can sort all DVDs in the cart by title or by cost:

- Sort by title: the system displays all the DVDs in the alphabet sequence by title. In case they have the same title, the DVDs having the higher cost will be displayed first.
- Sort by cost: the system the system displays all the DVDs in decreasing cost order. In case they have the same cost, the DVDs will be ordered by increasing title.

Customers can update the quantity of a DVD in a cart or remove a DVD from a cart. To increase consumer demand for the product and grow sales, customers are allowed to have an item for free which is randomly picked out in the cart by the system. Customers can filter DVDs in the cart by providing either its ID or title. If the item is found, display information of the found item in the cart. Or else, notify the customer the item is not found in the current cart.

The customer can request to place order when they are seeing the current cart. For simplification, he/she does not need to log in to place an order. The application will prompt the customer to enter the delivery information and delivery instructions. The software will then calculate the delivery fee based on the total mass of the order & the delivery location. Then, it will display to the customer the invoice including the DVD list, total cost before VAT, total cost after VAT, and the delivery fee. The customer can then proceed to pay for the order. Currently, only one payment method – i.e. credit card – is allowed by connecting to a card association system for checking the validation of the card or performing the pay transaction. After the transaction, the AIMS software will display all the detailed information such as transaction ID, card owner, transaction amount, transaction message, balance, transaction date to the customer. The order will be in pending state and the information of the order & the transaction will be sent to the customer's email.

2.2. Đối với Store Manager:

The store manager needs to log in to the system to navigate to the management mode. He/she can see the list of pending orders, then can pick any order to see its detail to approve or reject the order. The store manager can add new DVDs to the store. He or she must provide all information of the new DVD, including its ID, title, category, director, length, and the cost. Additionally, the manager can also remove DVDs from the store.

3. Use Case Diagram

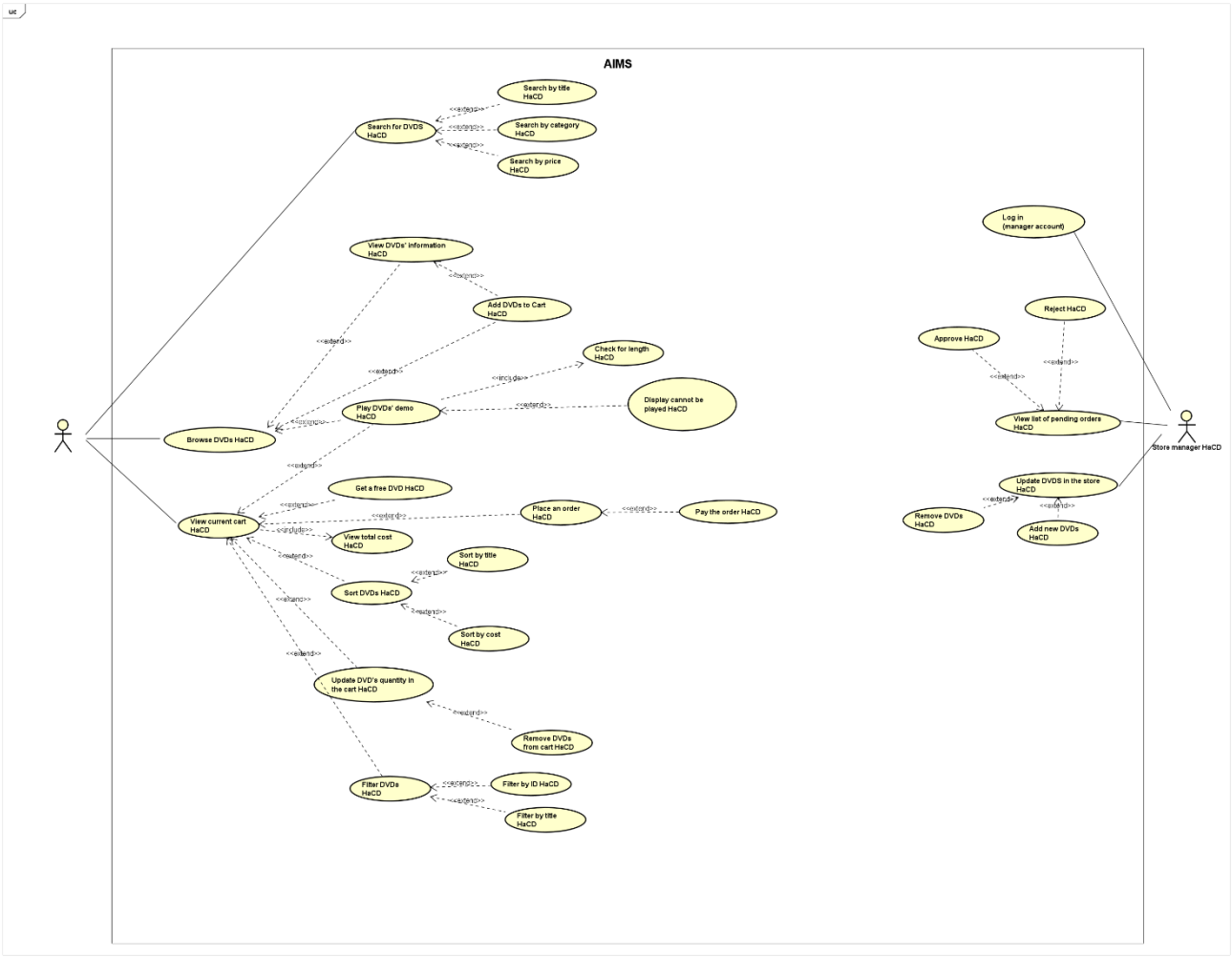

Figure 1: Use Case Diagram

4. Class Diagram

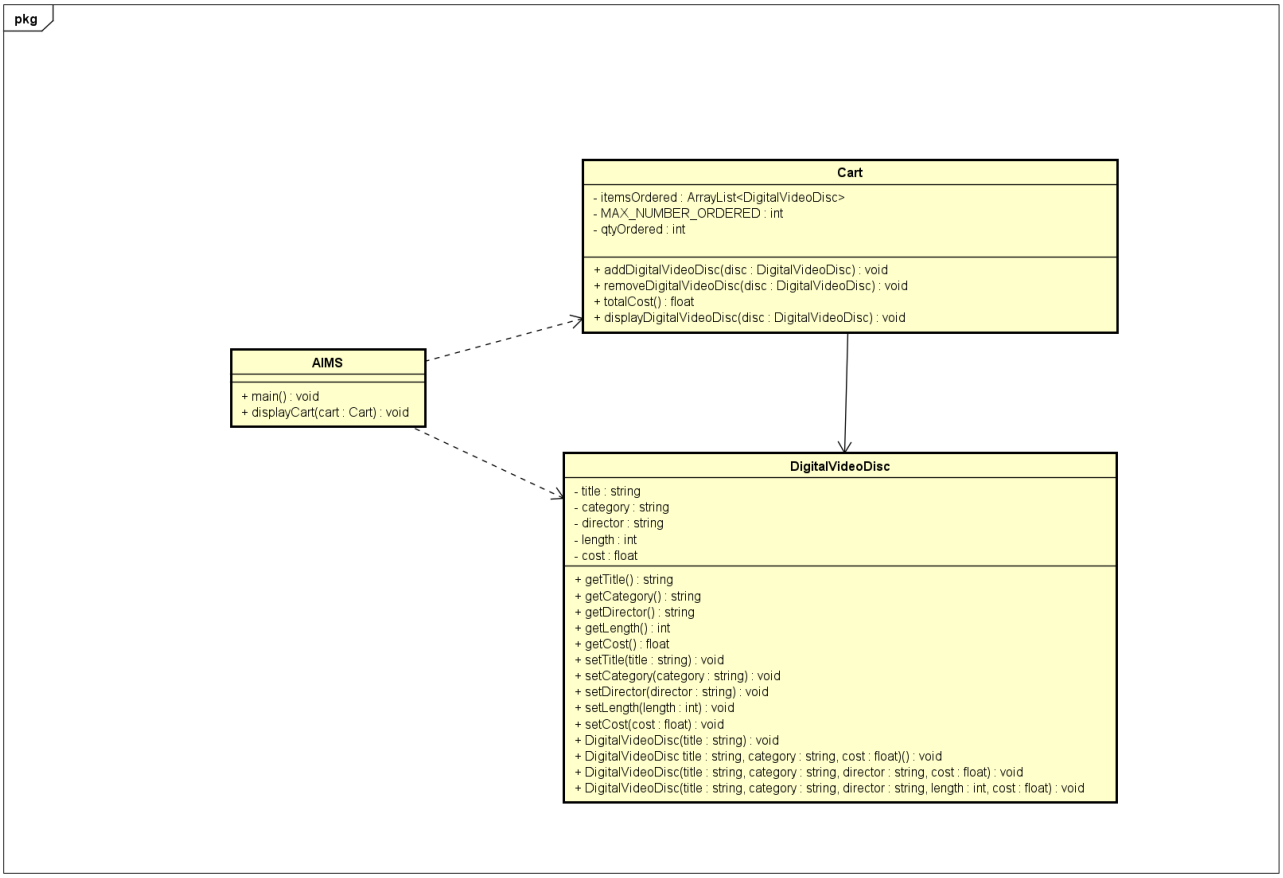

Figure 2: Class Diagram

5. Source Code

5.1. Aims Class

Aims.java

AimsProject > src > Aims.java > ...

```java

public class Aims {
    Run | Debug | Run main | Debug main
    public static void main(String[] args){
        //Create a new cart
        Cart anOrder = new Cart();

        //Create new dvd objects and add them to the cart
        DigitalVideoDisc dvd1 = new DigitalVideoDisc(title:"The Lion King", category:"Animation",director:"Roger Allers",
        anOrder.addDigitalVideoDisc(dvd1);

        DigitalVideoDisc dvd2 = new DigitalVideoDisc(title:"Star Wars", category:"Science Fiction", director:"George Lucas
        anOrder.addDigitalVideoDisc(dvd2);

        DigitalVideoDisc dvd3 = new DigitalVideoDisc(title:"Aladin", category:"Animation", cost:18.99f);
        anOrder.addDigitalVideoDisc(dvd3);

        //print total cost of the items in the cart
        System.out.println("Total cost is: " + anOrder.totalCost());


        //remove a dvd from the cart and print the new total cost
        anOrder.removeDigitalVideoDisc(dvd2);
        System.out.println("Total cost is: " + anOrder.totalCost());
    }
}

```

Figure 3: Aims Class

5.2. DigitalVideoDisc Class

DigitalVideoDisc.java

AimsProject > src > DigitalVideoDisc.java

```java
public class DigitalVideoDisc {
    private String title;
    private String category;
    private String director;
    private int length;
    private float cost;

    public String getTitle(){
        return title;
    }

    public String getCategory(){
        return category;
    }

    public String getDirector(){
        return director;
    }

    public int getLength(){
        return length;
    }

    public float getCost(){
        return cost;
    }
```

Figure 4: DigitalVideoDisc Class 1

```java

    public DigitalVideoDisc(String title,String category, float cost){
        super();
        this.title = title;
        this.category = category;
        this.cost = cost;
    }

    public DigitalVideoDisc(String title, String category, String director, float cost){
        super();
        this.title = title;
        this.category = category;
        this.director= director;
        this.cost = cost;
    }

    public DigitalVideoDisc(String title, String category, String director, int length, float cost){
        super();
        this.title = title;
        this.category = category;
        this.director = director;
        this.length = length;
        this.cost = cost;
    }

```

```
27
28      public void setTitle(String title){
29          this.title = title;
30      }
31
32      public void setCategory(String category){
33          this.category = category;
34      }
35
36      public void setDirector(String director){
37          this.director = director;
38      }
39
40      public void setLength(int length){
41          this.length = length;
42      }
43
44      public void setTitle(float cost){
45          this.cost = cost;
46      }
47
48      public DigitalVideoDisc(String title){
49          super();
50          this.title = title;
51      }
52
```

Figure 5: DigitalVideoDisc Class 2

Cart Class

Cart.java 1

AimsProject > src > Cart.java > Language Support for Java(TM) by Red Hat > Cart

```java
public class Cart {
    public static final int MAX_NUMBER_ORDERED = 20;
    private DigitalVideoDisc itemsOrdered[] = new DigitalVideoDisc[MAX_NUMBER_ORDERED];

    public int qtyOrdered;

    public void addDigitalVideoDisc(DigitalVideoDisc disc){
        if(qtyOrdered >= MAX_NUMBER_ORDERED){
            System.out.println(x:"The cart is almost full");
        }else{
            itemsOrdered[qtyOrdered] = disc;
            qtyOrdered++;
            System.out.println(x:"The disc has been added");
        }
    }


    public void removeDigitalVideoDisc(DigitalVideoDisc disc){
        int findDisc = -1;
        if(qtyOrdered > 0){
            for(int i = 0; i < qtyOrdered; i++){
                if(itemsOrdered[i].getTitle().equals((disc.getTitle()))){
                    findDisc = i;
                }
            }
            if(findDisc != -1){
                for(int i = findDisc; i < qtyOrdered - 1; i++){
                    itemsOrdered[i] = itemsOrdered[i + 1];
                }

```

Figure 6: Cart Class 1

```
30
31                  itemsOrdered[qtyOrdered - 1] = null;
32                  qtyOrdered--;
33
34                  System.out.println(x:"The disc has been removed");
35              }else{
36                  System.out.println(x:"The disc has not been found");
37              }
38          }
39      }
40
41
42
43
44      public float totalCost(){
45          float sum = 0;
46          for(int i = 0; i < qtyOrdered; i++){
47              sum += itemsOrdered[i].getCost();
48          }
49          return sum;
50      }
51
52  }
```

Figure 7: Cart Class 2

6. Kết quả demo

Aims.java

AimsProject > src > Aims.java > ...

```java

public class Aims {
    Run | Debug | Run main | Debug main
    public static void main(String[] args){
        //Create a new cart
        Cart anOrder = new Cart();

        //Create new dvd objects and add them to the cart
        DigitalVideoDisc dvd1 = new DigitalVideoDisc(title:"The Lion King", category:"Animation",director:"Roger Allers",
        anOrder.addDigitalVideoDisc(dvd1);

        DigitalVideoDisc dvd2 = new DigitalVideoDisc(title:"Star Wars", category:"Science Fiction", director:"George Lucas
        anOrder.addDigitalVideoDisc(dvd2);

        DigitalVideoDisc dvd3 = new DigitalVideoDisc(title:"Aladin", category:"Animation", cost:18.99f);
        anOrder.addDigitalVideoDisc(dvd3);

        //print total cost of the items in the cart
        System.out.println("Total cost is: " + anOrder.totalCost());


        //remove a dvd from the cart and print the new total cost
        anOrder.removeDigitalVideoDisc(dvd2);
        System.out.println("Total cost is: " + anOrder.totalCost());
    }
}

```

Figure 8: Code Demo

Kết quả:

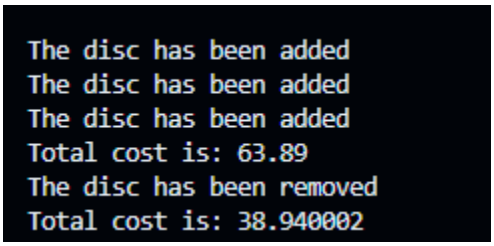

Figure 9: Result

7. Reading Assignment

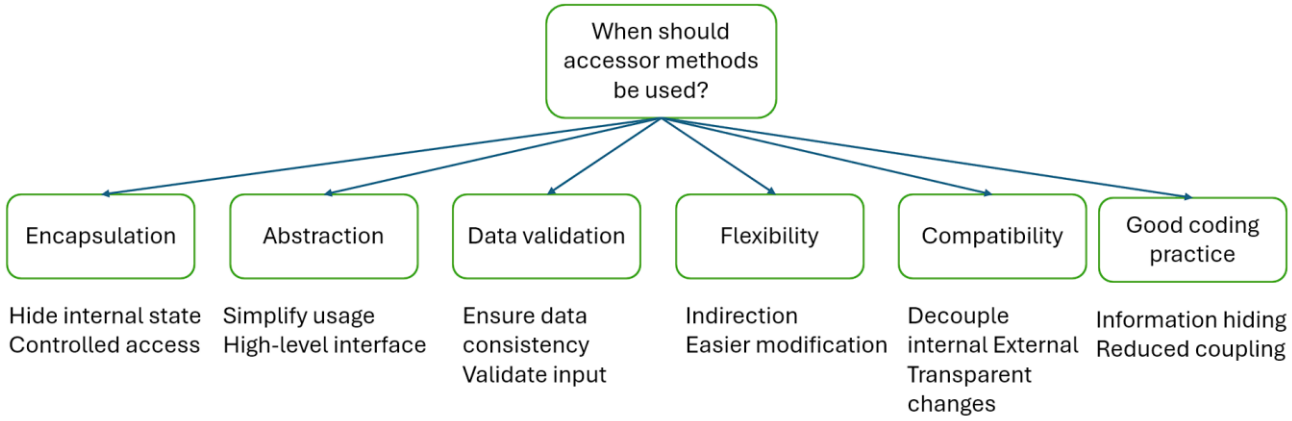

Figure 10: Getter and Setter Methods

8. Answer the question

Question: If you create a constructor method to build a DVD by title then create a constructor method to build a DVD by category. Does JAVA allow you to do this?
Answer: Java does not allow this because both constructors would have a single String parameter, which would cause ambiguity. If you create a constructor that accepts a String for the title and another constructor that also accepts a String for the category, Java cannot determine which constructor to call. To solve this problem, you could either:

1. Use a different parameter type apart from String for category.
2. Use a different parameter order if there are multile parameters.