**Design Pattern ver2.0**

Written by: GROUP 3 – 1 January 2025

Contents

[1. Class Cart 2](#_Toc186656135)

[2. Class CartMedia 2](#_Toc186656136)

[3. Class Media 2](#_Toc186656137)

[4. Class Order 3](#_Toc186656138)

[5. Class PaymentTransaction 3](#_Toc186656139)

# 1. Class Cart

|  |  |  |  |
| --- | --- | --- | --- |
| # | Pattern | Description | Improvement |
|  | Singleton Pattern | Có duy nhất 1 giỏ hàng được khởi tạo trong hệ thống | Đã triển khai |
|  | Iterator Pattern | Hiện tại, việc duyệt qua danh sách lstCartMedia sử dụng vòng lặp for. Nếu danh sách lớn hoặc cần hỗ trợ các loại duyệt khác nhau (forward, backward, etc.), | * Tạo interface CartIterator với các phương thức như hasNext() và next(). * Thêm một class CartMediaIterator để duyệt qua danh sách lstCartMedia. |
|  | Observer Pattern | Cần Gửi thông báo đến khách hàng khi giỏ hàng thay đổi. Cập nhật giao diện UI khi sản phẩm được thêm vào hoặc xóa. | * Tạo interface CartObserver với phương thức update(). * Class Cart sẽ giữ danh sách các observer (UI, Logger, etc.). * Khi thêm hoặc xóa sản phẩm, gọi notifyObservers() để thông báo thay đổi. |
|  | Strategy Pattern | Hiện tại, phương thức calSubtotal() đang tính tổng giá trị giỏ hàng bằng cách nhân giá từng sản phẩm với số lượng. Cần có thiết kế để áp dụng các chiến lược tính toán khác nhau (giảm giá, khuyến mãi, phí ship, v.v.) | * Tạo interface PricingStrategy với phương thức calculateTotal(). * Triển khai các chiến lược như DefaultPricingStrategy, DiscountPricingStrategy. * Thay thế logic trong calSubtotal() bằng cách sử dụng chiến lược được chọn. |

# 2. Class CartMedia

|  |  |  |  |
| --- | --- | --- | --- |
| # | Pattern | Description | Improvement |
| 2.1. | Factory Method Pattern | Hiện tại, việc khởi tạo CartMedia thông qua constructor với nhiều tham số (media, quantity, price) có thể gây khó khăn nếu logic khởi tạo thay đổi. | * Tạo một factory class CartMediaFactory để quản lý việc tạo CartMediaFactory * Sử dụng trong Cart |

# 3. Class Media

|  |  |  |  |
| --- | --- | --- | --- |
| # | Pattern | Description | Improvement |
| 3.1. | Factory Method Pattern | Hiện tại, việc khởi tạo Media có thể gây khó khăn vì cần phải biết chính xác loại của đối tượng. | * Tạo ra một Factory để sinh các đối tượng Media dựa trên thông tin được cung cấp. * Sử dụng Factory này khi tạo các đối tượng Media. |
| 3.2. | Observer Pattern | Cần gửi thông tin đến khách hàng khi số lượng Media thay đổi | * Tạo interface MediaObserver với phương thức update(). * Class Media sẽ giữ danh sách các observer (UI, Logger, etc.). * Khi thêm hoặc xóa sản phẩm, gọi notifyObservers() để thông báo thay đổi. |

# 4. Class Order

|  |  |  |  |
| --- | --- | --- | --- |
| # | Pattern | Description | Improvement |
| 3.1. | Observer Pattern | Cần gửi thông tin đến khách hàng khi thông tin về order thay đổi | * Tạo interface OrderObserver với phương thức update(). * Class Order sẽ giữ danh sách các observer (UI, Logger, etc.). * Khi thêm hoặc xóa sản phẩm, gọi notifyObservers() để thông báo thay đổi. |

# 5. Class PaymentTransaction

|  |  |  |  |
| --- | --- | --- | --- |
| # | Pattern | Description | Improvement |
| 3.1. | Observer Pattern | Cần gửi thông tin đến khách hàng khi thông tin về order thay đổi | * Tạo interface OrderObserver với phương thức update(). * Class Order sẽ giữ danh sách các observer (UI, Logger, etc.). * Khi thêm hoặc xóa sản phẩm, gọi notifyObservers() để thông báo thay đổi. |