

Lab 02: Factory method, Abstract factory

Áp dụng Factory Method và Abstract Factory trong phát triển phần mềm

Bài tập này giúp người học nắm bắt và thực hành kết hợp hai mẫu thiết kế phổ biến, áp dụng vào bài toán thực tế trong phát triển phần mềm, từ đó hiểu rõ hơn về nguyên lý thiết kế hướng đối tượng và các mẫu thiết kế.

1. Giới thiệu bài toán

Bạn được giao nhiệm vụ phát triển một hệ thống quản lý đơn hàng cho một công ty thương mại điện tử. Công ty này cung cấp nhiều loại sản phẩm khác nhau và các đơn hàng có thể đến từ nhiều kênh bán hàng như cửa hàng trực tuyến, cửa hàng vật lý, hoặc qua ứng dụng di động. Mỗi kênh bán hàng có thể có cách xử lý đơn hàng khác nhau, bao gồm cả cách tính toán phí vận chuyển, hình thức thanh toán và cách quản lý kho hàng.

Công ty mong muốn xây dựng một hệ thống có tính mở rộng, dễ bảo trì và có thể thêm các kênh bán hàng mới mà không làm ảnh hưởng đến các kênh hiện có.

2. Yêu cầu thiết kế và cài đặt

Yêu cầu thiết kế:

- Sử dụng Factory Method để tạo các đối tượng cụ thể cho các kênh bán hàng khác nhau.
- Sử dụng Abstract Factory để cung cấp các nhóm đối tượng có liên quan, chẳng hạn như đối tượng xử lý đơn hàng, đối tượng tính phí vận chuyển, và đối tượng thanh toán.
- Hệ thống phải hỗ trợ ít nhất ba kênh bán hàng: cửa hàng trực tuyến, cửa hàng vật lý, và ứng dụng di động.

Yêu cầu cài đặt:

- Xây dựng các lớp cụ thể cho từng kênh bán hàng, mỗi lớp cần có phương thức để xử lý đơn hàng và phương thức để tính phí vận chuyển.
- Xây dựng các lớp cụ thể để đại diện cho các hình thức thanh toán khác nhau như thẻ tín dụng, thanh toán khi nhận hàng (COD), và thanh toán qua ví điện tử.
- Thiết kế và triển khai Abstract Factory để tạo ra các đối tượng phù hợp dựa trên loại kênh bán hàng.

3. Gợi ý hướng dẫn giải quyết vấn đề

Factory Method:

- Tạo một lớp cơ sở hoặc giao diện cho các kênh bán hàng (`SalesChannel`).
- Mỗi kênh bán hàng cụ thể (cửa hàng trực tuyến, cửa hàng vật lý, ứng dụng di động) sẽ là một lớp con cụ thể, kế thừa từ lớp cơ sở hoặc giao diện.
- Xác định một phương thức `createOrderProcessor()` trừu tượng trong lớp cơ sở để tạo các đối tượng xử lý đơn hàng, và các lớp con sẽ cài đặt phương thức này.

Abstract Factory:

- Tạo một giao diện hoặc lớp trừu tượng `OrderFactory` với các phương thức tạo đối tượng cần thiết như `createShippingCalculator()`, `createPaymentProcessor()`.

- Cài đặt các lớp con cụ thể của `OrderFactory` để cung cấp các đối tượng xử lý phù hợp cho từng kênh bán hàng.
- Sử dụng các lớp Abstract Factory trong phương thức xử lý đơn hàng của các kênh bán hàng cụ thể.

4. Kết quả cần đạt

- Hệ thống được cài đặt phải đáp ứng được yêu cầu đề ra, có thể dễ dàng thêm mới các kênh bán hàng hoặc phương thức thanh toán mà không cần thay đổi cấu trúc hiện tại.
- Các lớp và phương thức phải được thiết kế sao cho có tính tái sử dụng và mở rộng cao.

5. Hướng phát triển mở rộng bài toán

- Thêm kênh bán hàng mới như qua mạng xã hội hoặc qua các đối tác liên kết.
- Mở rộng hệ thống để hỗ trợ các loại sản phẩm đặc thù cần quy trình xử lý riêng.
- Tích hợp thêm các phương thức thanh toán mới hoặc các cách tính phí vận chuyển phức tạp hơn.

6. Hướng dẫn đánh giá

- Kiểm tra tính đúng đắn: Xem xét việc triển khai có tuân thủ đúng các nguyên tắc của Factory Method và Abstract Factory không.
- Kiểm tra tính mở rộng: Đánh giá khả năng thêm mới các kênh bán hàng hoặc phương thức thanh toán mà không ảnh hưởng đến hệ thống hiện tại.
- Kiểm tra tính tái sử dụng: Các lớp và phương thức có thể tái sử dụng trong các tình huống tương tự khác không.
- Kiểm tra tài liệu: Xem xét sự rõ ràng và chi tiết trong tài liệu hướng dẫn cách sử dụng hệ thống.