

Design for class “OrderController”

Table 1. Example of attribute design

#	Name	Data type	Default value	Description
1	orderList	OrderList	null	Danh sách đơn hàng
2	order	Order	null	Đơn hàng cụ thể cần xử lý

Table 2. Example of operation design

#	Name	Return type	Description (purpose)
1	requestOrderList(OrderList orderList)	void	Yêu cầu danh sách đơn hàng
2	returnsListOfOrders()	OrderList	Trả về danh sách các đơn hàng
3	submitOrderCancellation(order_id)	Order	Gửi yêu cầu hủy đơn hàng dựa trên order_id

Parameter

- `orderList`: Danh sách các đơn hàng.
- `order_id`: Mã đơn hàng cần hủy.

Exception

- `OrderNotFoundException`: Nếu không tìm thấy đơn hàng theo `order_id`.
- `InvalidOrderException`: Nếu đơn hàng không thể hủy do trạng thái hiện tại.

Method

- `requestOrderList(OrderList orderList)`: Gửi yêu cầu lấy danh sách đơn hàng.
- `returnsListOfOrders()`: Trả về danh sách các đơn hàng hiện có.
- `submitOrderCancellation(order_id)`: Hủy đơn hàng dựa vào `order_id`.

How to use parameters / attributes

- `orderList` chứa danh sách đơn hàng để xử lý.
- `order_id` được truyền vào để xác định đơn hàng cần hủy.

Flowchart / Activity diagram / Sequence diagram

- Nếu cần, có thể tạo biểu đồ **sequence diagram** mô tả cách OrderUI tương tác với OrderController và Order.

State

- **Pending** → Khi đơn hàng đang chờ xử lý.
- **Processed** → Khi đơn hàng đã được xử lý thành công.

- **Cancelled** → Khi đơn hàng bị hủy.

State diagram (if any)

- Có thể vẽ sơ đồ trạng thái để biểu diễn quá trình thay đổi trạng thái của đơn hàng.

Design for class “OrderHistoryUI”

Table 1. Example of attribute design

#	Name	Data type	Default value	Description
1	orderList	OrderList	null	Danh sách các đơn hàng của người dùng
2	order	Order	null	Đơn hàng cụ thể đang được hiển thị hoặc xử lý

Table 2. Example of operation design

#	Name	Return type	Description (purpose)
1	goToOrderHistoryPage()	void	Điều hướng đến trang lịch sử đơn hàng
2	displayOrderListOnInterface()	void	Hiển thị danh sách đơn hàng trên giao diện
3	selectOrderToCancel(OrderList, order_id)	Order	Chọn đơn hàng cần hủy dựa vào order_id
4	displayCancellationConfirmationMessage()	void	Hiển thị thông báo xác nhận hủy đơn hàng
5	confirmCancellation()	void	Xác nhận hủy đơn hàng đã chọn

Parameter

- `orderList`: Danh sách các đơn hàng của người dùng.
- `order_id`: Mã đơn hàng cần hủy.

Exception

- `OrderNotFoundException`: Nếu không tìm thấy đơn hàng theo `order_id`.
- `UIException`: Nếu giao diện gặp lỗi khi hiển thị danh sách đơn hàng.

Method

- `goToOrderHistoryPage()`: Chuyển hướng đến trang lịch sử đơn hàng.

- `displayOrderListOnInterface()`: Hiển thị danh sách đơn hàng trên giao diện.
- `selectOrderToCancel(OrderList, order_id)`: Chọn đơn hàng cần hủy dựa trên `order_id`.
- `displayCancellationConfirmationMessage()`: Hiển thị thông báo xác nhận hủy đơn hàng.
- `confirmCancellation()`: Xác nhận hủy đơn hàng.

How to use parameters / attributes

- `orderList` chứa danh sách đơn hàng được hiển thị trên giao diện.
- `order_id` được truyền vào để xác định đơn hàng cần hủy.

Flowchart / Activity diagram / Sequence diagram

- Có thể vẽ **sequence diagram** thể hiện quá trình người dùng chọn đơn hàng để hủy và xác nhận hủy.

State

- **Viewing Order List** → Khi người dùng đang xem danh sách đơn hàng.
- **Selecting Order to Cancel** → Khi người dùng chọn một đơn hàng để hủy.
- **Cancellation Confirmed** → Khi đơn hàng đã được xác nhận hủy.

State diagram (if any)

- Có thể vẽ sơ đồ trạng thái để mô tả sự thay đổi trạng thái của giao diện người dùng.

Design for class “Order”

Table 1. Example of attribute design

#	Name	Data type	Default value	Description
1	orderList	OrderList	null	Danh sách đơn hàng lấy từ hệ thống
2	totalCost	double	0.0	Tổng chi phí của đơn hàng

Table 2. Example of operation design

#	Name	Return type	Description (purpose)
1	getOrderListFromSystem(OrderList)	void	Lấy danh sách đơn hàng từ hệ thống
2	requestToCalculateTotalCost(order)	void	Yêu cầu tính tổng chi phí của một đơn hàng

Parameter

- `orderList`: Danh sách đơn hàng trong hệ thống.
- `order`: Đối tượng đơn hàng cần tính tổng chi phí.

Exception

- `OrderNotFoundException`: Nếu không tìm thấy đơn hàng trong hệ thống.
- `CalculationException`: Nếu có lỗi khi tính tổng chi phí đơn hàng.

Method

- `getOrderListFromSystem(OrderList)`: Truy xuất danh sách đơn hàng từ hệ thống.
- `requestToCalculateTotalCost(order)`: Yêu cầu tính tổng chi phí của đơn hàng.

How to use parameters / attributes

- `orderList` chứa tất cả các đơn hàng của người dùng.
- `totalCost` sẽ được tính toán và cập nhật khi gọi phương thức `requestToCalculateTotalCost(order)`.

Flowchart / Activity diagram / Sequence diagram

- Có thể vẽ **sequence diagram** thể hiện quy trình lấy danh sách đơn hàng và tính tổng chi phí.

State

- **Pending Calculation** → Khi đơn hàng chưa được tính tổng chi phí.
- **Total Cost Calculated** → Khi tổng chi phí đã được tính thành công.

State diagram (if any)

- Có thể vẽ sơ đồ trạng thái để mô tả sự thay đổi trạng thái của đơn hàng trong hệ thống.

