

# POINT OF SALE TERMINAL SYSTEM

20181577 김경준

## 목차

1. Vision & Scope
2. Use Case
3. Design Refinement
4. Interaction Diagram
5. Class Diagram
6. Result

## 1. Vision & Scope

### Vision

옷가게를 위한 POST 시스템이다. 본래 POST시스템의 역할인 고객 응대시간 단축과 매출관리의 용이성을 목표로 한다. 또한, 옷가게는 재고가 존재하며 그 변동이 자주 일어나므로, 이 POST시스템을 이용하여 재고관리를 쉽게 만들고 재고변동을 파악할 수 있게 한다.

### 이 제품의 당사자들

가게 사장 : 재고를 관리하고 매출을 정리하며, 가게 전반을 관리하고 매출에 따라 사업 목표를 설정한다.

판매 직원 : POST기를 사용하여 물건을 판매하고 고객의 요구사항을 들어준다.

고객 : 물건을 구매할 때 직원을 통하여 여러가지 구매에 필요한 참고사항을 얻는다.

개발자 : 시스템을 개발하고 유지/보수 한다.

### Scope

기본 POST 시스템에서 요구되는 기능

- 상품 등록/제거
- 할인율 적용
- 매출 확인
- 영수증 출력
- 환불

추가되는 기능

- 재고 관리(추가, 변경, 검색)

## 2. Use Case

### Use Case Description

Use case name	상품 등록
Scenario	상품의 바코드를 스캔하여 구매할 목록에 상품을 등록한다.
Triggering event	1. 고객이 원하는 상품을 가져온다 2. 새로운 재고가 들어온다.
Brief description	고객이 상품들을 들고 오거나 새로운 재고가 들어오면, 상품의 바코드를 스캔하거나 상품 번호를 입력하여 상품 목록에 등록한다.
Actors	점원, 상품 목록
Related use cases	없다
Preconditions	등록할 상품이 존재한다.
Post conditions	상품이 등록된다.
Flow of activities	1. 상품의 바코드를 스캔하거나 상품번호를 입력한다.  2. 상품을 상품 목록에 등록한다.
Exception conditions	할인이 진행중인 상품이면 할인 적용 use-case를 실행한다.

Use case name	재고 변경
Scenario	상품을 구매하거나, 재고 변경이 필요할 때
Triggering event	1. 상품을 구매한다. 2. 새로운 재고가 들어온다.

	3. 재고 DB와 실제 재고에 차이가 발생한다.	
Brief description	재고의 변경이 필요할 때, 재고DB와 상호작용 하여 입력값 만큼의 재고를 변경한다.	
Actors	상품 목록, 재고 DB	
Related use cases	상품 구매, 상품 등록	
Preconditions	1. 상품 목록에 변경할 만큼의 상품을 등록한다. 2. 상품 구매를 진행한다.	
Post conditions	재고가 변경된다.	
Flow of activities	시스템	
	1. 재고를 추가할지 줄일지 정한다.  2. 상품 목록만큼의 재고를 재고DB값에서 변경한다.	
Exception conditions	1. DB 접근에 실패하면, 접근 실패 오류 메시지를 출력한다. 2. 재고를 줄일 때, 줄이려는 값이 DB의 값보다 큰 경우, 재고가 없다는 오류 메시지를 출력한다.	
Use case name	상품 구매	
Scenario	결제	
Triggering event	고객이 원하는 모든 상품의 등록과 변동사항을 끝마친다.	
Brief description	상품 목록의 모든 상품의 가격을 더한 후, 고객에게서 현금이나 카드로 결제한다. 그 후, 재고를 변동하고 영수증을 출력한다.	
Actors	점원, 고객, 상품 목록	
Related use cases	영수증 출력, 재고 변경	
Preconditions	상품 목록에 상품이 등록되어 있어야 한다..	
Post conditions	영수증 출력, 재고 변경, 상품 목록 초기화	
Flow of activities	점원	시스템
	1. 점원이 상품 구매를 선택한다.  2. 고객이 현금이나 카드로 결제한다.	1.1 상품 목록의 상품의 가격을 모두 더한 값을 제시한다.  3. 결제 이후 상품 목록에 등록된 만큼의 재고를 줄인다.  4. 구매가 완료된 시각과 고객의 구매정보가 입력된 영수증을 출력한다.

		5. 판매내역 DB에 판매내역을 등록한다.  6. 상품 목록을 초기화한다.
Exception conditions	1. 재고 변경에 실패 했을 경우, 상품구매 이전상태로 초기화 한다.	

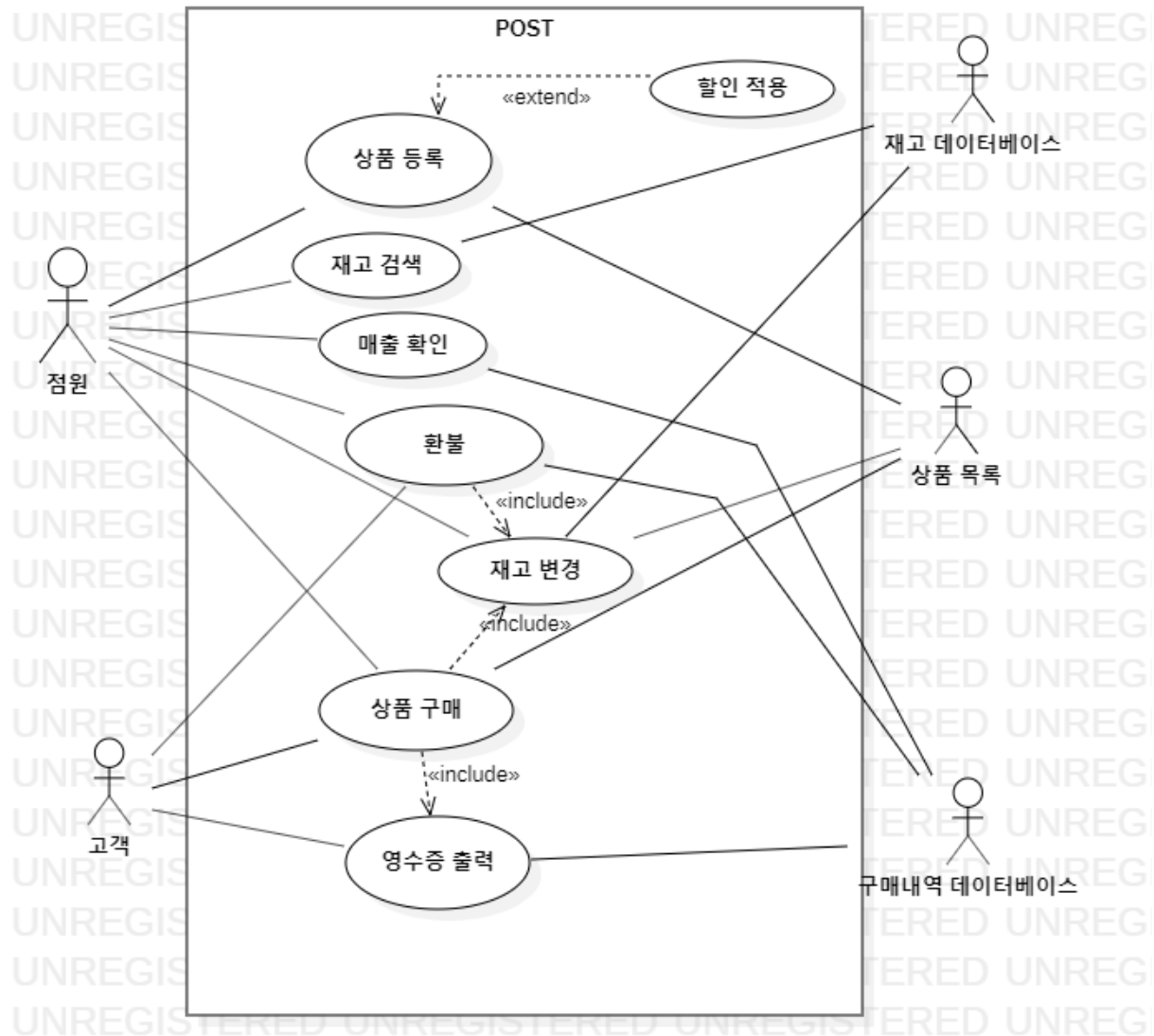
Use case name	할인 적용
Scenario	할인이 적용중이거나 상품권을 가진 고객의 상품을 가격을 줄인다.
Triggering event	현재 해당 상품의 할인이 진행중이다. 고객이 상품권을 사용한다.
Brief description	등록된 상품을 선택하고, 할인 비율이나 금액을 선택한다.
Actors	점원, 상품 목록
Related use cases	상품 등록
Preconditions	할인을 적용할 상품을 등록한다.
Post conditions	상품의 가격이 할인된다.
Flow of activities	1. 상품의 바코드를 스캔하거나 상품번호를 입력한다.  2. 상품을 상품 목록에 등록한다.
Exception conditions	전체 상품의 가격이 할인 적용 가격보다 작은 경우, 상품 가격이 0 원 이하로 떨어지지 않도록 한다.

Use case name	매출 확인
Scenario	오늘과 이번 달의 매출을 확인한다.

Triggering event	이번 달의 매출을 확인한다.
Brief description	판매 내역 DB에서 날짜별로 판매한 내역의 금액을 모두 더한 뒤 화면에 출력한다.
Actors	점원, 판매 내역 DB
Related use cases	없다.
Preconditions	없다.
Post conditions	없다.
Flow of activities	1. 판매 내역 DB에서 오늘 판매한 내역의 금액을 모두 더한다. 2. 화면에 출력한다.
Exception conditions	없다.

Use case name	환불	
Scenario	고객의 환불 요청	
Triggering event	제품의 하자나 변심으로 고객이 환불을 요구한다.	
Brief description	제품의 하자나 변심으로 고객이 환불을 하면, 판매 내역을 확인한 후, 지운다.	
Actors	고객, 점원, 판매 내역 DB, 재고 DB	
Related use cases	재고 변경	
Preconditions	고객이 환불을 요청한다. 이미 구매한 상품이다.	
Post conditions	판매 내역이 삭제되고 재고가 증가한다.	
Flow of activities	액터	시스템
	1. 환불 정책에 맞는 조건인지 확인한 후, 환불을 선택하여 영수증 번호를 입력한다.	1.1 영수증 번호에 대응하는 판매 내역을 찾는다.
	2. 구매 당시의 결제 조건에 따라 결제 취소나 현금을 전달한다.	2.1 취소가 완료되면, 결제 취소를 출력하고 판매 내역 DB의 해당 판매내역을 결제 취소 상태로 바꾼다.
Exception conditions	1. 영수증 번호에 대응하는 판매 내역이 없을 경우, 환불을 취소한다.	

## Use Case Diagram



## Requirements

기능적 요구사항	내용	중요도
재고 등록/삭제	관리자가 상품의 재고를 변경한다.	3/5
재고 검색	관리자가 상품을 검색하면 상태와 개수를 확인한다.	3/5
상품 등록/삭제	관리자가 상품번호를 입력하면 구매할 내용에 상품내역을 변경한다.	5/5

상품 구매	고객이 결제하면 구매할 내역의 상품의 매출과 재고감소를 반영한다.	5/5
영수증 출력	상품 구매가 완료되면 상품 구매내역을 텍스트 파일로 정리한다.	4/5
환불	상품 구매내역을 확인하여 그 구매내역에 대한 환불을 진행한다. 상품의 매출과 재고를 반영한다.	2/5
매출 확인	오늘 하루동안 판매된 매출을 확인한다.	1/5
할인 적용	상품을 선택하여 할인을 제공한다.	1/5

비기능적 요구사항	내용	품질 속성과 내용	중요도
빠른 반응	물건 구매를 완료하고 영수증 파일이 출력되기까지 시간을 단축시킨다.	Efficiency - 2초 이내에 모든 동작이 완료되어야 한다.	3/5
직관적 ui	ui구성을 그래픽으로 하고 기능별로 구분하여 원하는 기능 탐색이 가능하도록 한다.	Usability - 별다른 학습 없이 ui만으로도 주어진 기능 파악이 가능해야 한다.	5/5
오류 최소화	상품의 부재나 재고 이상의 구매가 되지 않도록 한다.	Reliability - 예상되는 오류는 차단하고, 오류가 발생해도 시스템을 유지하도록 한다.	5/5
구매기록 보안	상품 구매 후 남겨진 기록이 다른 사용자가 볼 수 없도록 한다.	Functionality - 관리자 외에 구매기록을 볼 수 없도록 한다.	3/5

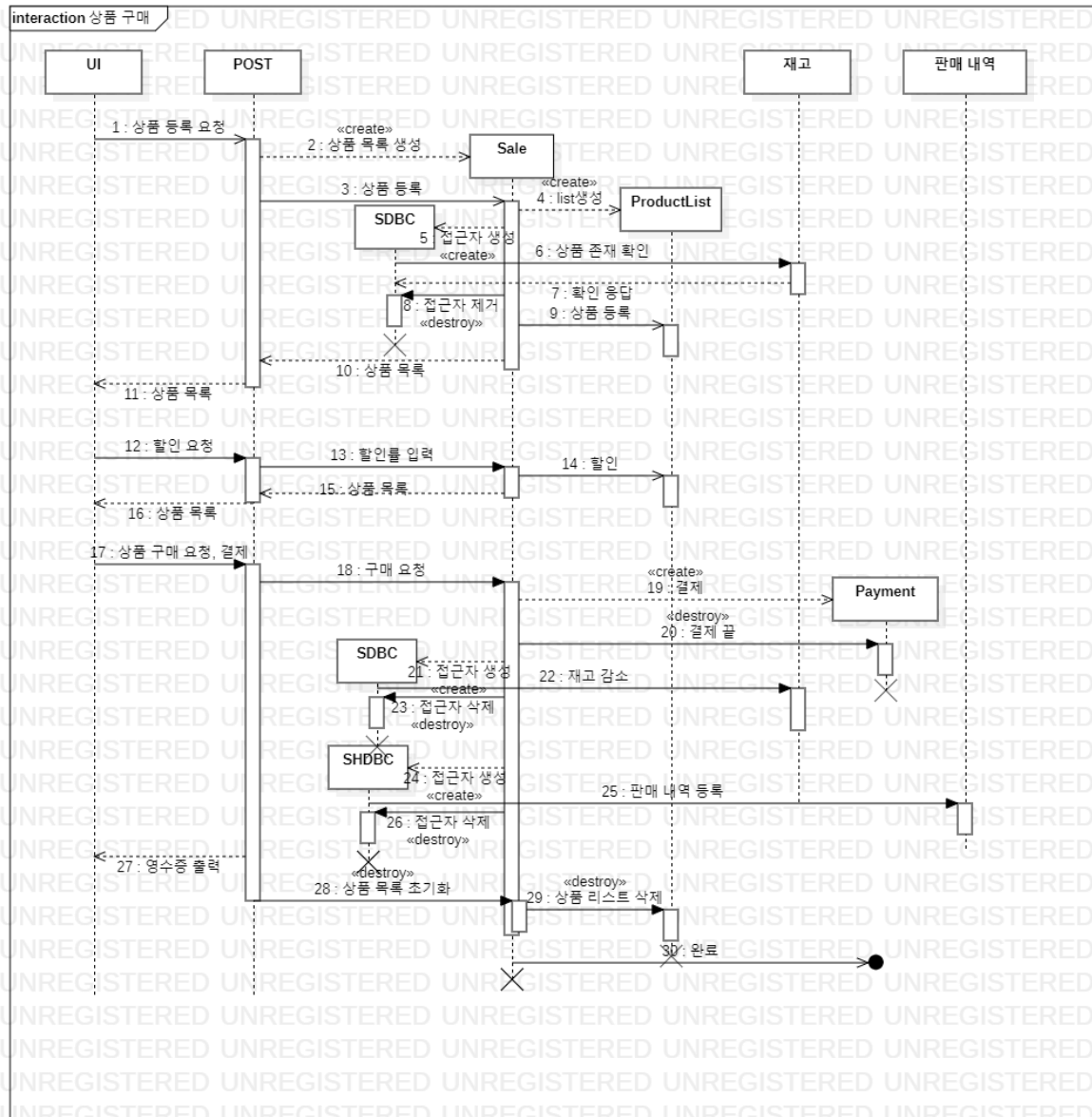
### 3. Design Refinement by GRASP principle and Design Patterns

Before 단계의 설계 classifier	After 단계의 설계 classifier	적용한 설계 개념	Architecture Design Rationale	기타
POST	POST, Sale, Refund, Bank	Expert, Controller	모든 일을 POST에서 처리하기 때문에 POST를 Controller 역할로만 사용하고 각각 등록/구매, 환불, 판매 총액을 수행하기 위한 class로 분화했습니다.	
-	Payment	Expert	Payment만을 수행하는 클래스로 분화했습니다.	
	PaymentInterface	Low coupling	구매 후, 재고와 판매내역 DB에 접근해야하는 클래스가 두 개 존재하여 복잡하게 묶이는 것을 줄이고 재사용성을 높이기 위해서 Interface를 만들었습니다.	

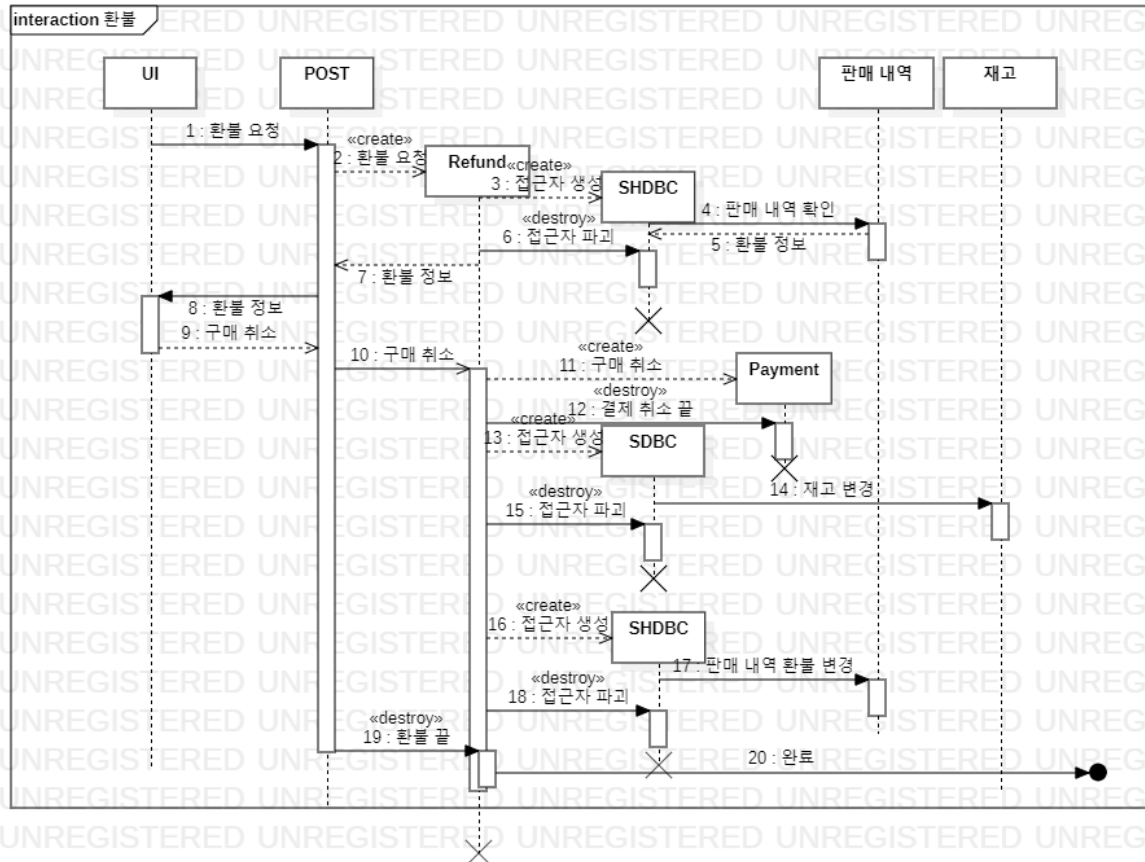


## 4. Interaction Diagram

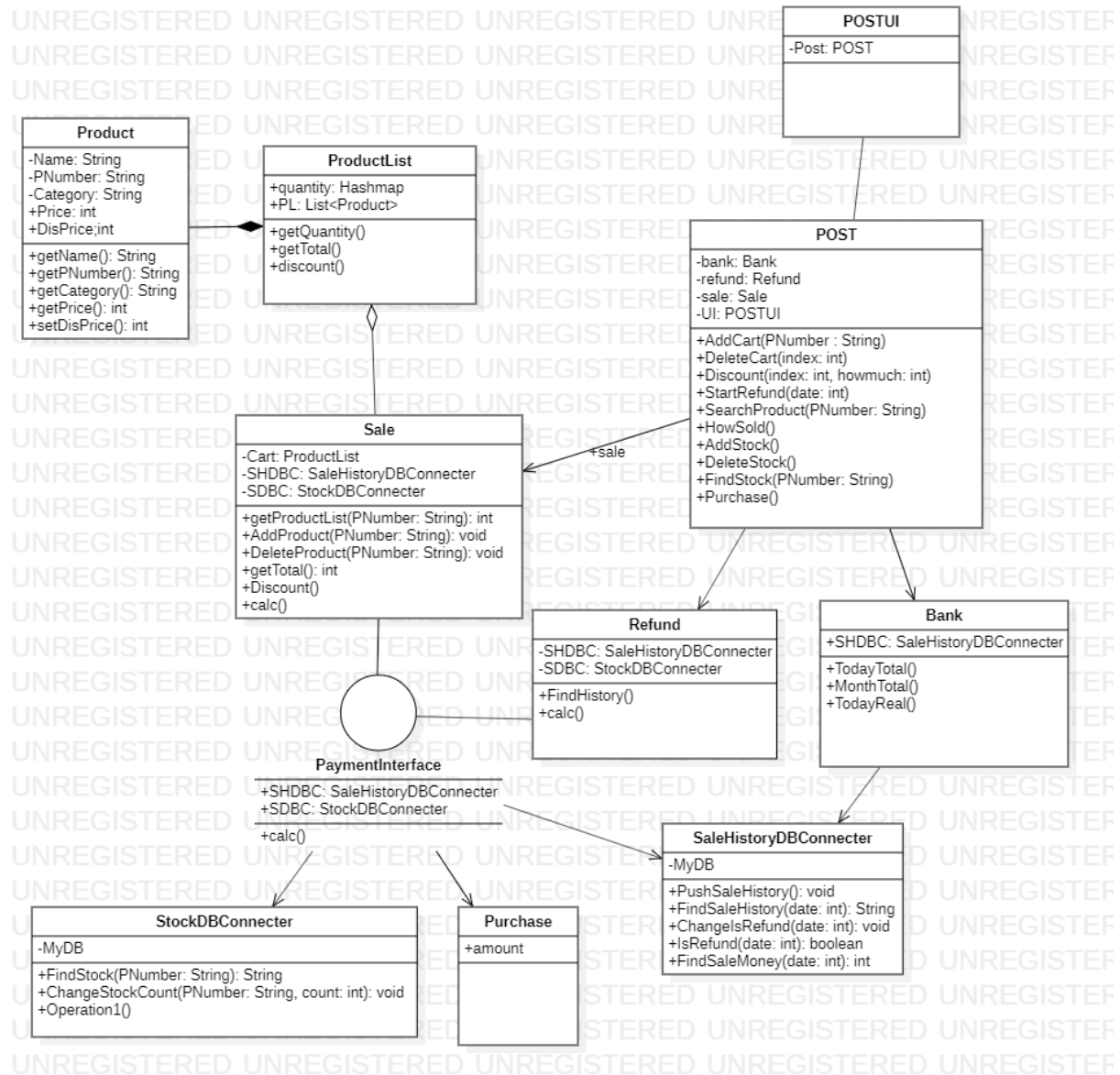
### 상품 구매 Interaction



## 환불 Interaction Diagram



## 5. Class Diagram



## 6.Result

JUn			
상품명	상품번호	정가	구매가격
sneaker	av123	30000	30000
jean	ab123	40000	32000
할인/제거		환불	결제
		매출	