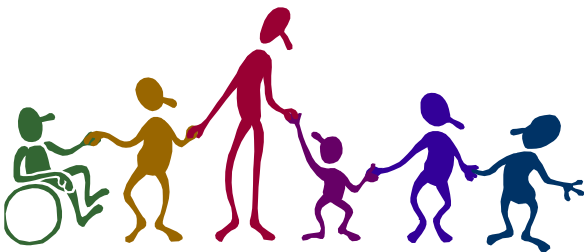


# 2장. 유스케이스 다이어그램



조 은 속  
서일대학교  
소프트웨어공학과  
escho@seoil.ac.kr

# 목차

- ➡ 유스케이스 개념
- ➡ 유스케이스 관계
- ➡ 유스케이스 모델링
- ➡ 유스케이스 다이어그램 실습

# 유스케이스 개요

- ➔ 유스케이스의 역할은 사용자의 시점에서 시스템을 모델링하는 것이다.
- ➔ 유스케이스는 시스템에 대한 시나리오의 집합이다.
- ➔ 각 시나리오에는 발생하는 이벤트의 흐름이 있다.
- ➔ 각 이벤트의 흐름은 사람, 시스템, 하드웨어, 혹은 시간의 흐름에 의해 시작된다.
- ➔ 이벤트 흐름을 시작하게 하는 객체가 액터이다.
- ➔ 이 흐름은 해당 액터 혹은 다른 액터에게 어떤 의미를 가지는 결과를 제공한다.

# 유스케이스 표기법



Actor



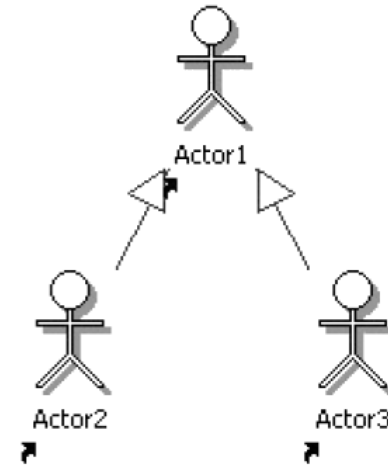
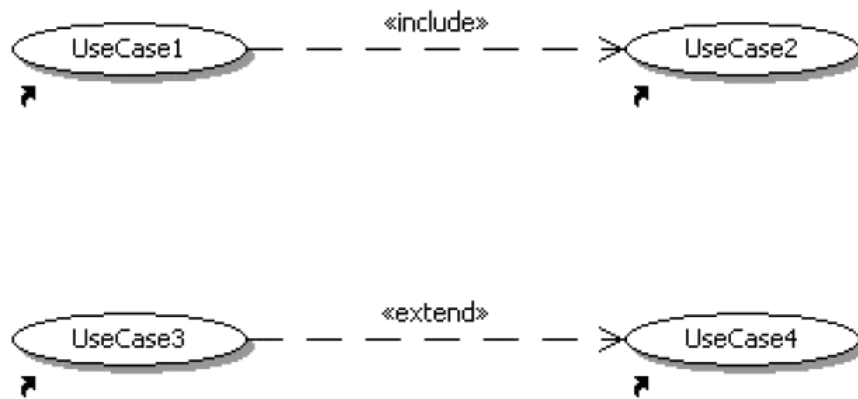
[그림 3-2] 유스케이스와 액터

# 유스케이스 관계

## ⇒ 연관관계

- ❑ 포함(Include)관계: 다른 유스케이스에서 기존의 유스케이스를 재사용할 수 있는 관계를 나타낸다.
- ❑ 확장(Extend)관계: 기존의 유스케이스에 진행단계를 추가하여 새로운 유스케이스를 만들어내는 관계이다.
- ❑ 일반화 관계: 추가로 액터와 유스케이스에 대한 일반화관계가 있을 수 있다.

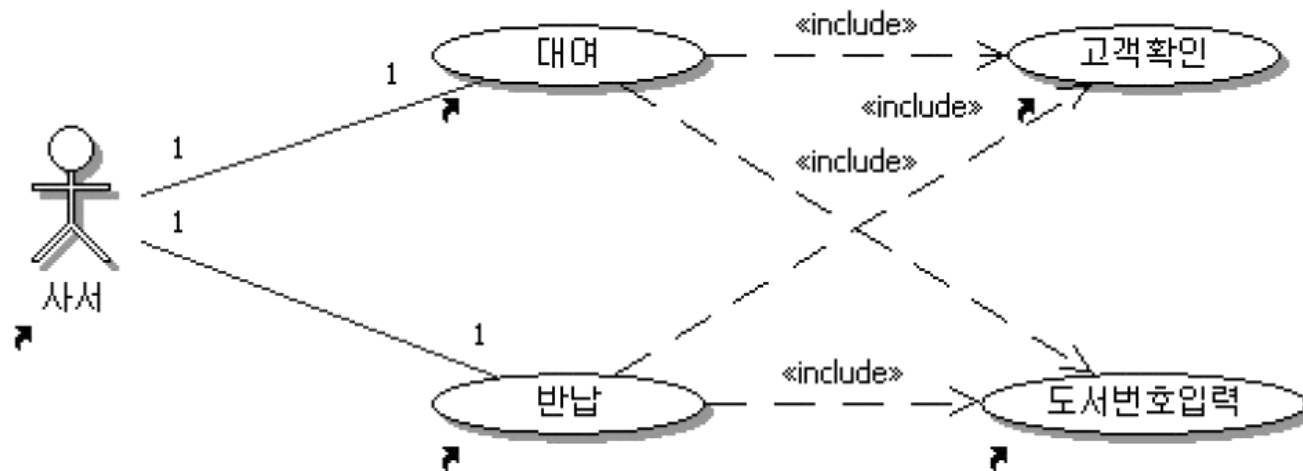
# 유스케이스 관계



[그림 3-3] 유스케이스 다이어그램에서 사용되는 관계들

# 포함 관계(Include)

- ➡ 하나의 유스케이스를 수행할 때, 같은 기능을 가진 다른 유스케이스가 반드시 수행되는 것을 말한다



[그림 3-4] 도서관 시스템의 포함관계

# 포함 관계(Include)

## ➔ 포함 관계에서의 이벤트 흐름

- ❑ 다음은 ‘대여’ 유스케이스와 ‘반납’ 유스케이스가 ‘고객확인’ 유스케이스와 도서번호 입력’ 유스케이스를 이용할 때의 이벤트 흐름을 보여준다.

[표 3-1] 포함관계를 이용한 ‘대여’와 ‘반납’ 유스케이스의 이벤트 흐름

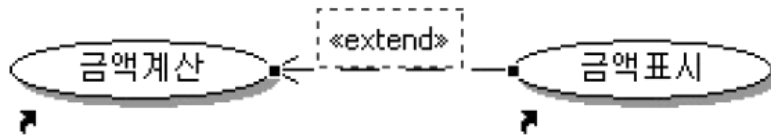
대여 유스케이스 이벤트 흐름	반납 유스케이스 이벤트 흐름
<ol style="list-style-type: none"><li>1. ‘고객확인’ 유스케이스를 포함한다.</li><li>2. 사서는 ‘대여’를 선택한다.</li><li>3. ‘도서번호입력’ 유스케이스를 포함한다.</li><li>4. 도서관 시스템은 고객이 대여가 가능한지 확인한다.</li><li>5. 고객에게 대여를 가능 여부를 표시한다.</li><li>6. 도서관 시스템은 도서를 대여 처리한다.</li></ol>	<ol style="list-style-type: none"><li>1. ‘고객확인’ 유스케이스를 포함한다.</li><li>2. 사서는 ‘반납’을 선택한다.</li><li>3. ‘도서번호입력’ 유스케이스를 포함한다.</li><li>4. 도서관 시스템은 도서를 반납 처리한다.</li></ol>



# 확장관계(Extends)

- ➡ 확장관계의 유스케이스는 포함관계처럼 여러 유스케이스에 걸쳐 중복적으로 사용되지 않고, 특정 조건에서 한 유스케이스로만 확장되는 것을 의미한다.
- ➡ 따라서 확장(extend)하는 유스케이스는 상위 유스케이스로부터 어떠한 특정 조건에 의해 수행됨을 의미한다.
- ➡ 포함관계가 여러 유스케이스에서 공통적으로 발견되는 시스템의 기능을 표현하기 위하여 사용되는 반면에 확장관계는 한 유스케이스에서 추가되거나 확장된 기능을 표현하기 위하여 사용된다.

# 확장 관계(Extend)



[그림 3-5] 금액계산 유스케이스의 확장

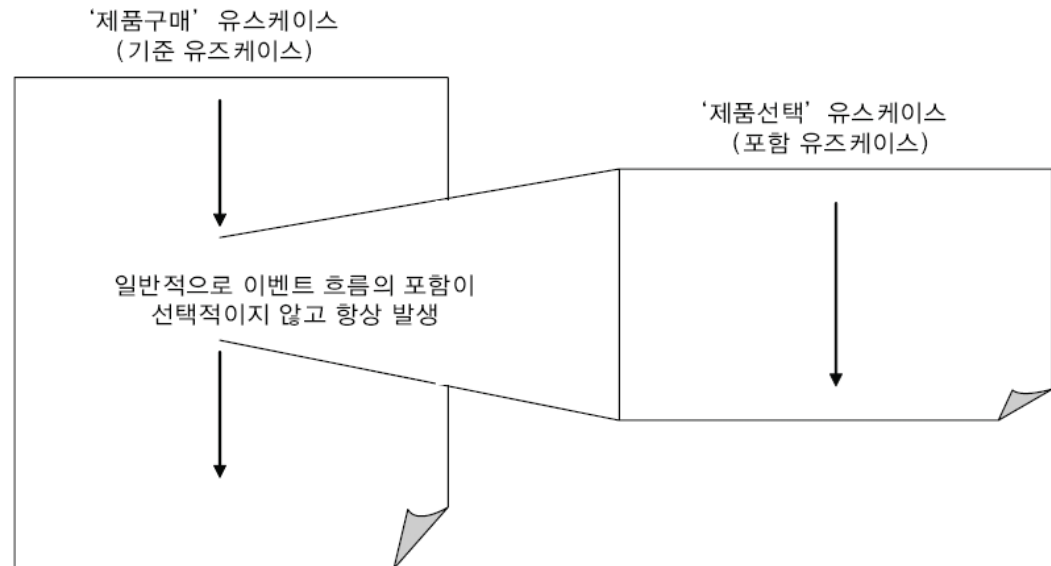
[표 3-2] 금액계산 유스케이스와 금액표시 유스케이스의 이벤트 흐름

금액계산 유스케이스 이벤트 흐름	금액표시 유스케이스 이벤트 흐름
<ol style="list-style-type: none"> <li>1. 사용자가 자판기에 동전을 투입하거나 음료수를 선택한다.</li> <li>2. 현재 금액에 투입된 동전만큼 액수를 추가한다.</li> <li>3. 금액표시 유스케이스 확장</li> </ol>	<ol style="list-style-type: none"> <li>1. 금액계산 유스케이스로부터 금액을 받는다.</li> <li>2. 받은 금액을 표시한다.</li> </ol>

# 유스케이스 관계

## ➡ 확장 유스케이스와 포함 유스케이스

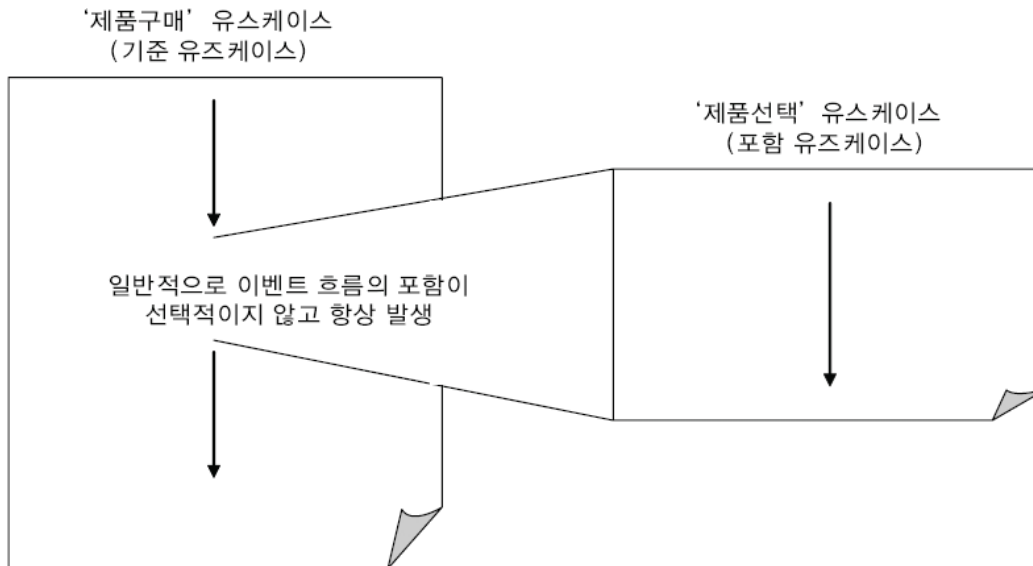
- 확장관계에 있는 유스케이스 사이의 이벤트 흐름은 포함관계에 있는 유스케이스 사이의 이벤트 흐름과 유사하다. 즉 기존 유스케이스의 이벤트 흐름이 수행되었다가 확장점을 만나면 지정된 유스케이스의 이벤트 흐름으로 분기된다.



[그림 3-8] 포함관계의 이벤트 흐름

# 유스케이스 관계

## ➡ 확장 유스케이스와 포함 유스케이스



[그림 3-8] 포함관계의 이벤트 흐름

# 유스케이스 관계

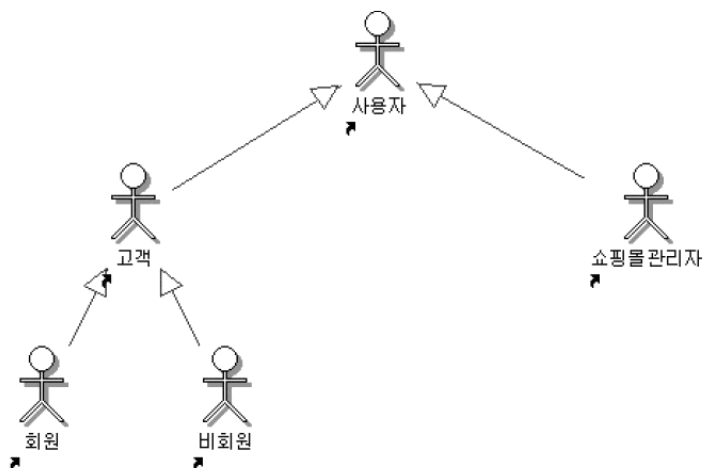
[표 3-3] 포함관계와 확장관계의 비교

	포함관계	확장관계
목 적	<ul style="list-style-type: none"><li>여러 유스케이스에 공통적인 기능을 표현하기 위해 사용된다.</li></ul>	<ul style="list-style-type: none"><li>기준 유스케이스에 부가적으로 추가된 기능을 표현하기 위해 사용된다.</li></ul>
이벤트 흐름	<ul style="list-style-type: none"><li>포함된 유스케이스로의 이벤트 흐름 분기가 필수적이다.</li><li>기준 유스케이스 이후의 이벤트 흐름이 포함된 유스케이스의 수행 결과에 의존한다.</li></ul>	<ul style="list-style-type: none"><li>확장 유스케이스는 확장 유스케이스에 기술된 조건에 따라 선택적으로 수행된다.</li><li>기준 유스케이스 이후의 이벤트 흐름이 확장 유스케이스의 결과에 의존하지 않는다.</li></ul>

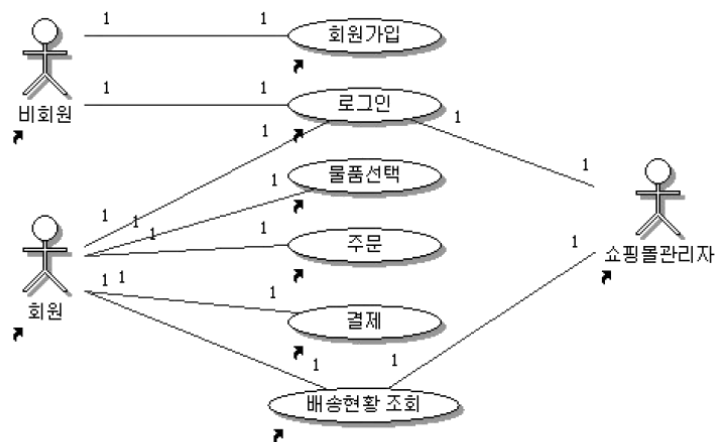
# 액터 관계

## ➡ 액터 사이의 일반화(Generalization)

- ❑ 추상적인 액터와 좀더 구체적인 액터 사이에 맺어주는 관계로서 “한 액터가 다른 액터의 일종이다” 또는 “한 액터도 다른 액터에 해당된다”라는 의미가 만족되는 두 액터를 일반화관계로 연결한다.



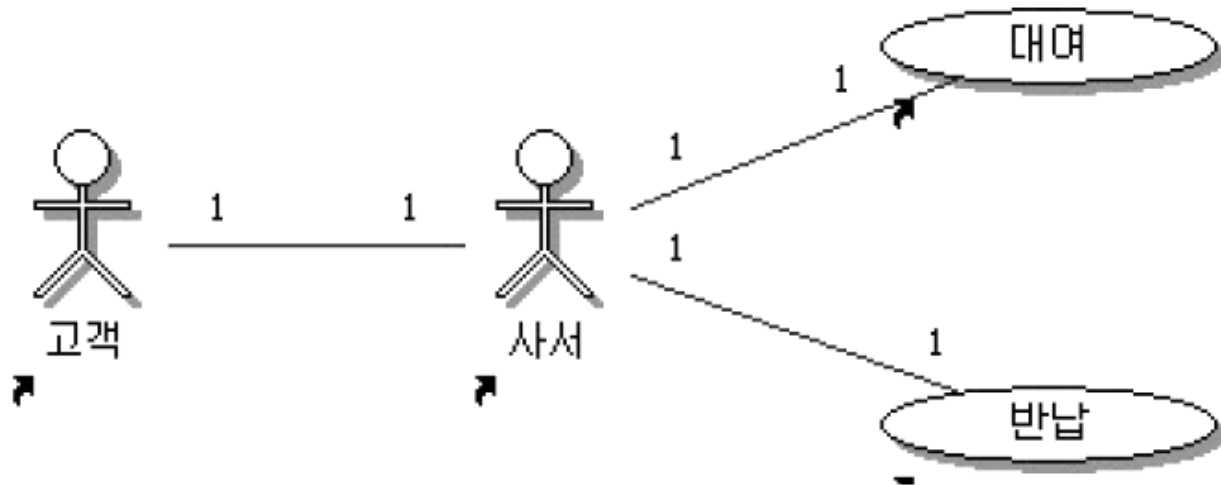
[그림 3-9] 쇼핑몰 액터들 사이의 일반화관계



[그림 3-10] 일반화를 사용하지 않은 경우

# 액터 관계

- ➡ 유스케이스 모델링 과정에서 2개 이상의 유스케이스 이벤트 흐름에서 중복적인 부분이 발생할 수 있다. 이러한 경우에 유스케이스 간 포함관계를 설정하여 해결한다.
- ➡ 도서관 시스템의 예



[그림 3-13] 도서관 시스템의 간략한 유스케이스 다이어그램

[표 3-4] 대여 및 반납 유스케이스의 이벤트 흐름

대여 유스케이스 이벤트 흐름	반납 유스케이스 이벤트 흐름
1. 사서는 고객의 아이디와 패스워드를 도서관 시스템에 입력한다.	
2. 도서관 시스템은 아이디와 패스워드로 고객의 정보를 확인하고 업무 종류를 선택할 수 있는 화면을 보여준다.	
3. 사서는 '대여'를 선택한다.	3. 사서는 '반납'을 선택한다.
4. 도서관 시스템은 사서에게 도서 번호를 입력하기 위한 화면을 보여준다.	
5. 사서는 도서 번호를 입력한다.	
6. 도서관 시스템은 도서 목록에서 도서 번호가 유효한지 확인한다.	
7. 도서관 시스템은 고객이 대여가 가능한지 확인한다.	7. 도서관 시스템은 도서를 반납 처리한다.
8. 고객에 대한 대여의 가능 여부를 표시한다.	
9. 도서관 시스템은 도서를 대여 처리한다.	



[표 3-5] '고객확인' 유스케이스와 '도서번호입력' 유스케이스

<p>고객확인</p>	<ol style="list-style-type: none"> <li>1. 사서는 고객의 아이디와 패스워드를 도서관 시스템에 입력한다.</li> <li>2. 도서관 시스템은 아이디와 패스워드로 고객의 정보를 확인하고 업무 종류를 선택할 수 있는 화면을 보여준다.</li> </ol>
<p>도서번호입력</p>	<ol style="list-style-type: none"> <li>1. 도서관 시스템은 사서에게 도서의 번호를 입력하기 위한 화면을 보여준다.</li> <li>2. 사서는 도서 번호를 입력한다.</li> <li>3. 도서관 시스템은 도서 목록에서 도서 번호가 유효한지 확인한다.</li> </ol>

# 유스케이스 모델링 절차

⇒ 1단계: 시스템 상황 확인

⇒ 2단계

- 액터 식별
- 행위자와 그들의 책임을 확인

⇒ 3단계

- 유스케이스 식별
- 특정한 목적의 관점에서 볼 때 쓰임새와 시스템의 특성 확인

⇒ 4단계

- 유스케이스 다이어그램 작성
- 행위자와 유스케이스에서 정제할 부분이 있는지 평가
- 유스케이스에서 <<include>> 의존성이 있는지 평가
- 유스케이스에서 <<extend>> 의존성이 있는지 평가
- 행위자와 유스케이스를 일반화(또는 공유)할 수 있는지 평가

# 유스케이스 모델링 절차

## ⇒ 5단계

- 유스케이스 명세서 작성
- 유스케이스명, 액터명 및 개요 기술
- 사전 및 사후 조건과 제약사항 식별
- 작업(정상, 대치, 예외)흐름과 시나리오 도출
- 유스케이스 흐름에서 포함이나 확장 유스케이스로 구조화

## ⇒ 6단계: 유스케이스 실체화

# 유스케이스 모델링 절차

## ➡ 1단계: 시스템 상황 분석(문제 기술서 작성)

- ❑ 입고관리 기능
- ❑ 출고관리 기능
- ❑ 현황관리 기능

## ➡ 2단계: 액터 식별



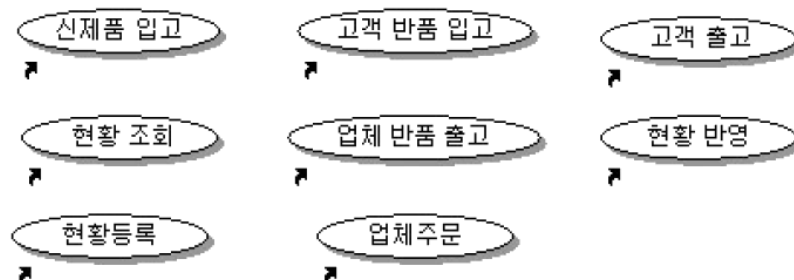
[그림 3-20] 재고관리 시스템의 액터

# 유스케이스 모델링 절차

## ➡ 3단계: 유스케이스 식별

[표 3-7] 재고관리 시스템의 사용자별 기능

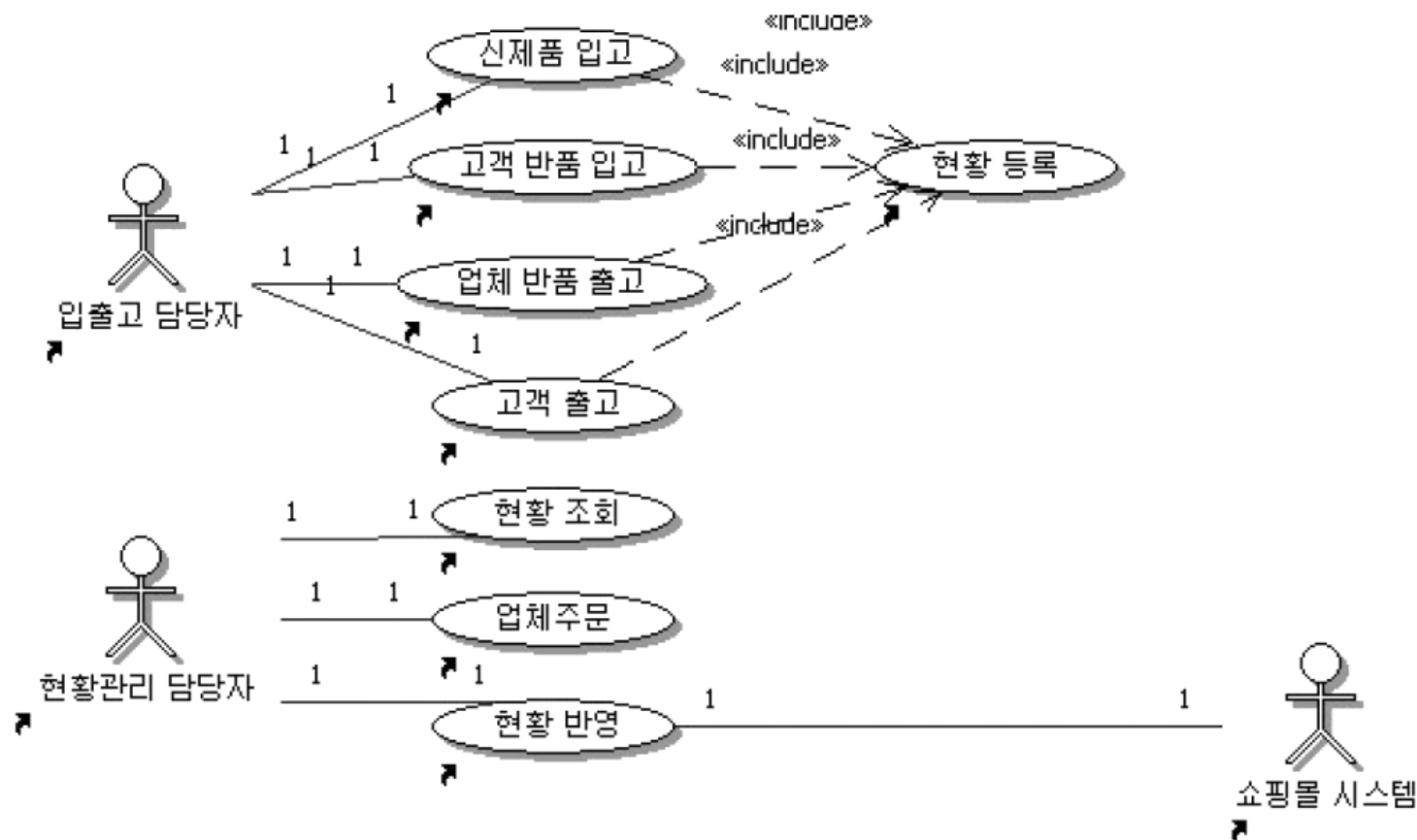
기능 범주	사용자	기능(유스케이스)
입고관리	입출고 담당자	통신업체 제품 입고 기능 고객 반품 입고 기능
출고관리	입출고 담당자	통신업체 반품 출고 기능 고객 출고 기능 발주 기능
현황관리	입출고 담당자	현황 조회
	현황관리 담당자	현재 현황 등록 기능(업데이트) 통신업체 주문 기능
	쇼핑몰 시스템	실시간 현황 반영 기능



[그림 3-21] 재고관리 시스템의 유스케이스

# 유스케이스 모델링 절차

## ➡ 4단계: 유스케이스 다이어그램 작성



[그림 3-22] 재고관리 시스템의 유스케이스 다이어그램

# 유스케이스 모델링 절차

## ➔ 5단계: 유스케이스 명세서 작성

- ▶ 유스케이스명 : 신제품 입고
- ▶ 액터명 : 입출고 담당자
- ▶ 유스케이스 개요 및 설명 : 입출고 담당자는 신제품이 입고되면 제품의 상태를 확인하고 입고 또는 반품시킨다.
- ▶ 사전 조건 : 현황관리 담당자가 통신업체에 주문한 제품이다.

### □ 이벤트 흐름

#### – 정상 흐름

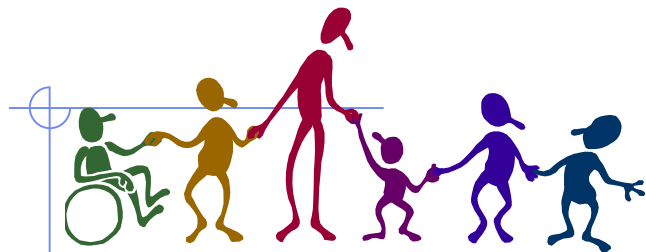
- ① 통신업체로부터 제품 입고를 요구한다.
- ② 입출고 담당자는 제품이 주문한 제품인지 확인한다.
- ③ 제품의 상태를 파악한다.
- ④ 제품을 입고하고 제품 목록을 현황관리 담당자에게 알린다.

#### – 선택 흐름

- ① 제품에 하자가 발생하면 현황관리 담당자에게 하자를 알리고 반품한다.

- ➔ 유스케이스:
- ➔ 포함관계:
- ➔ 확장관계:
- ➔ 액터의 의미:
- ➔ 액터의 식별:





## Q & A

