

# Use case Modeling

---



# Contents

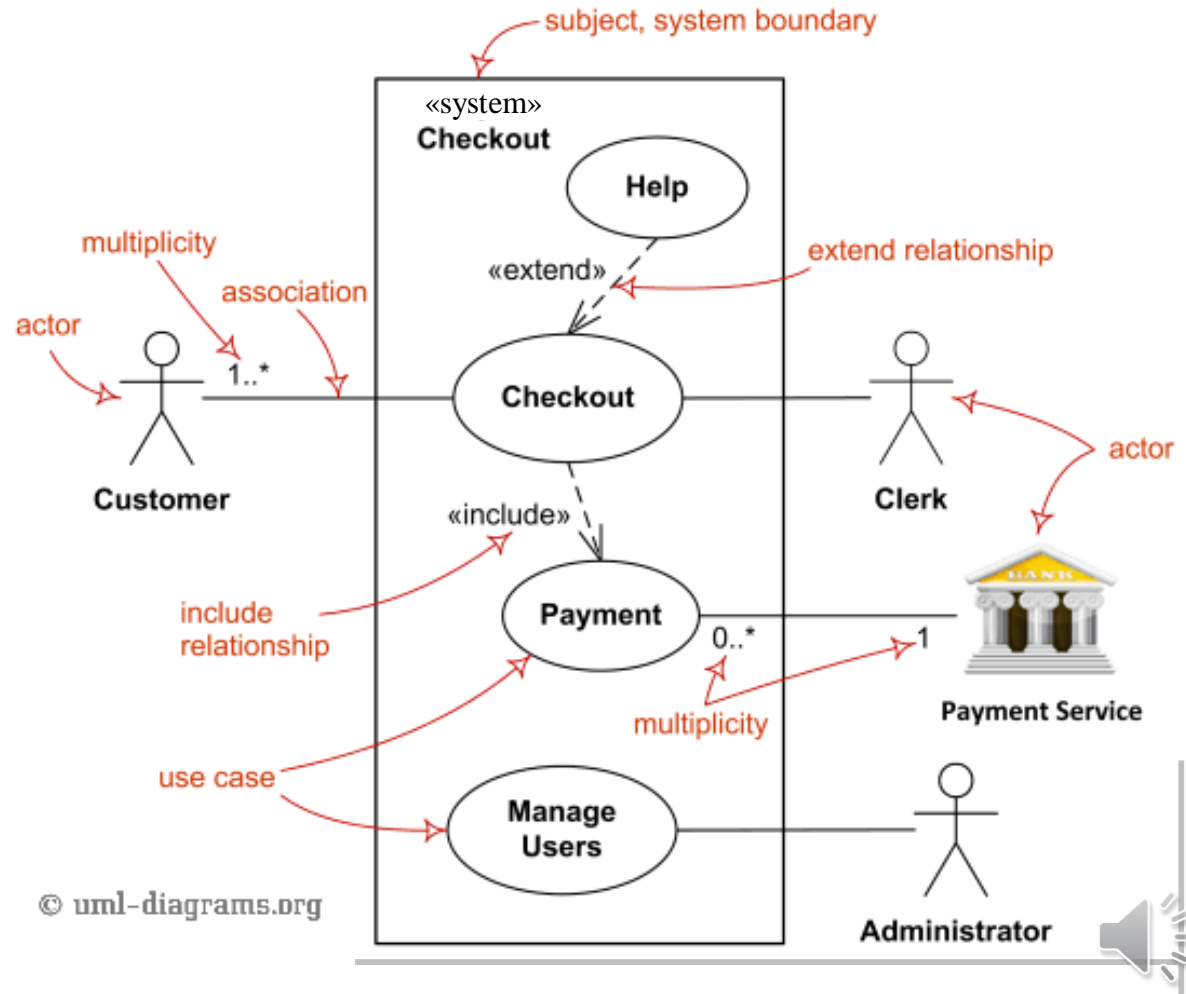
---

- 액터
- 유스케이스
- 액터와 유스케이스간의 관계
- 유스케이스 명세서



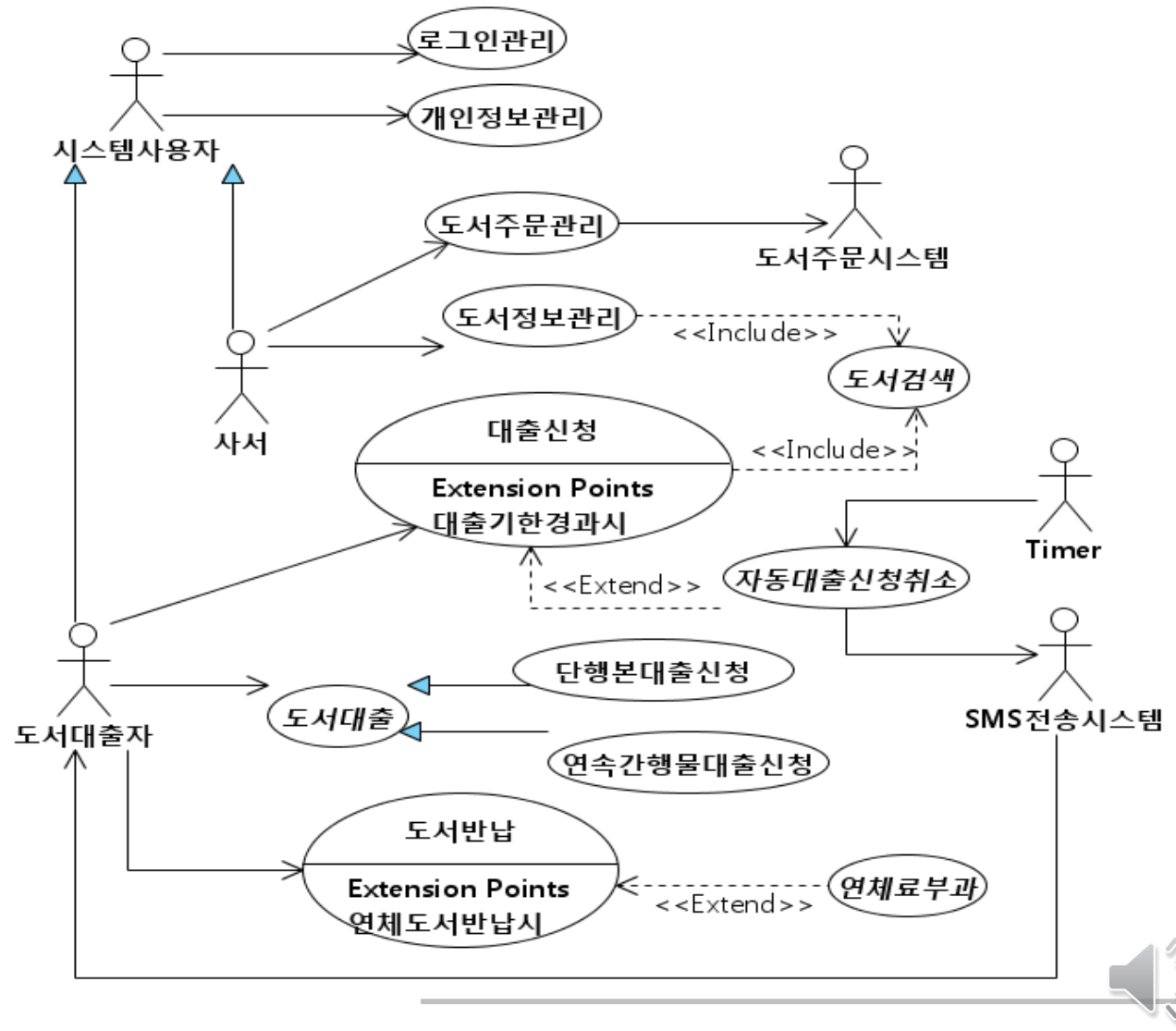
# Use case Diagram

- Use case diagrams used to describe a set of actions (use cases) that a system(subject) should perform with external entities of the system (actors).



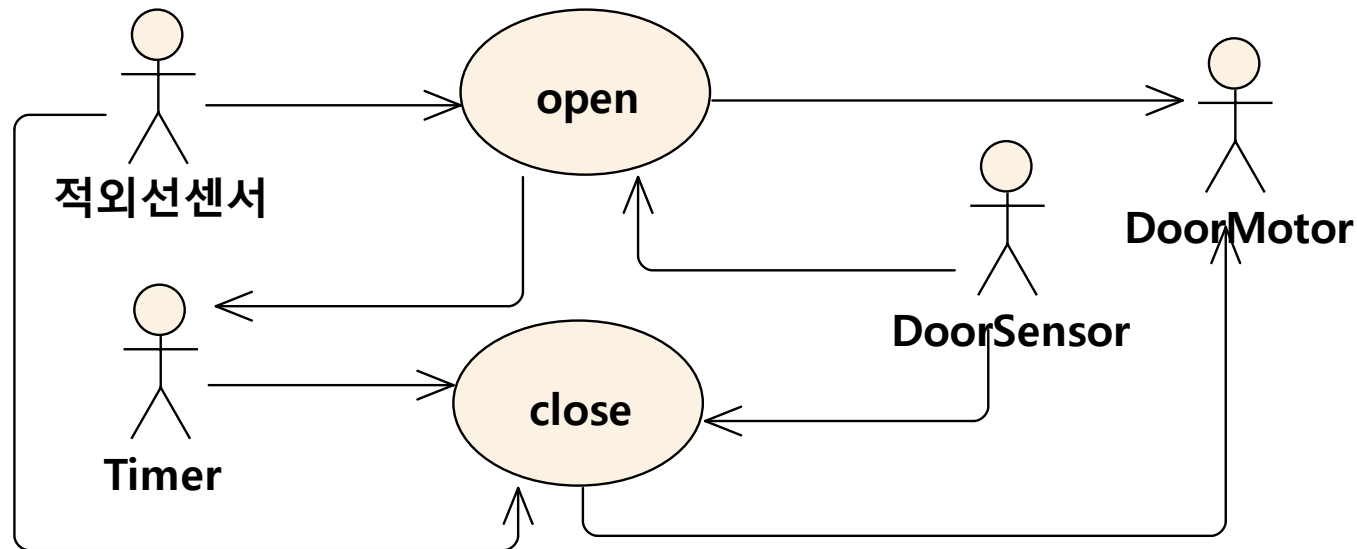
# Use case Model: Example

## ■ 도서관리 시스템



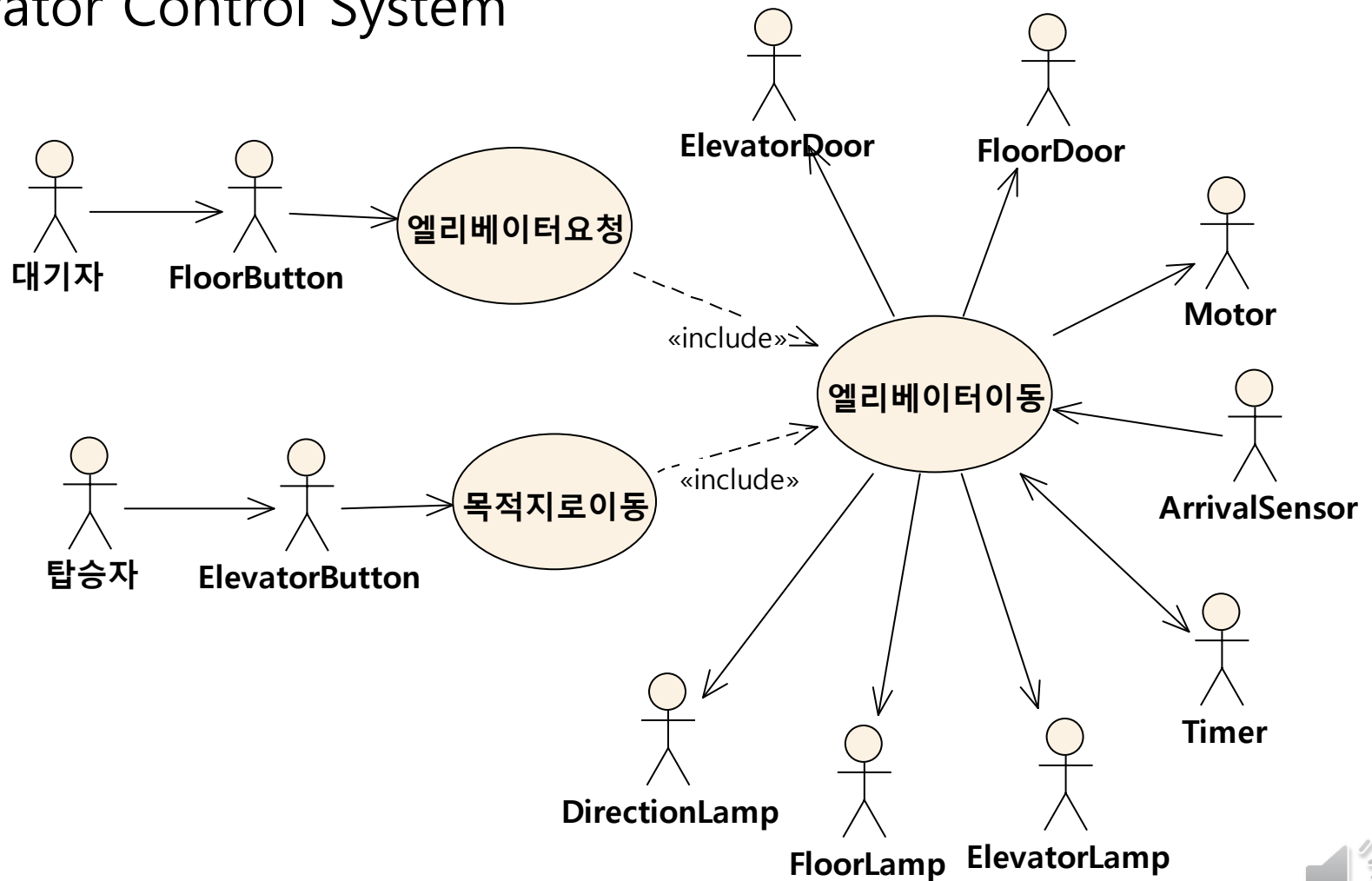
# Use Case Model - Example

## ■ 자동문



# Use Case Model - Example

## ■ Elevator Control System



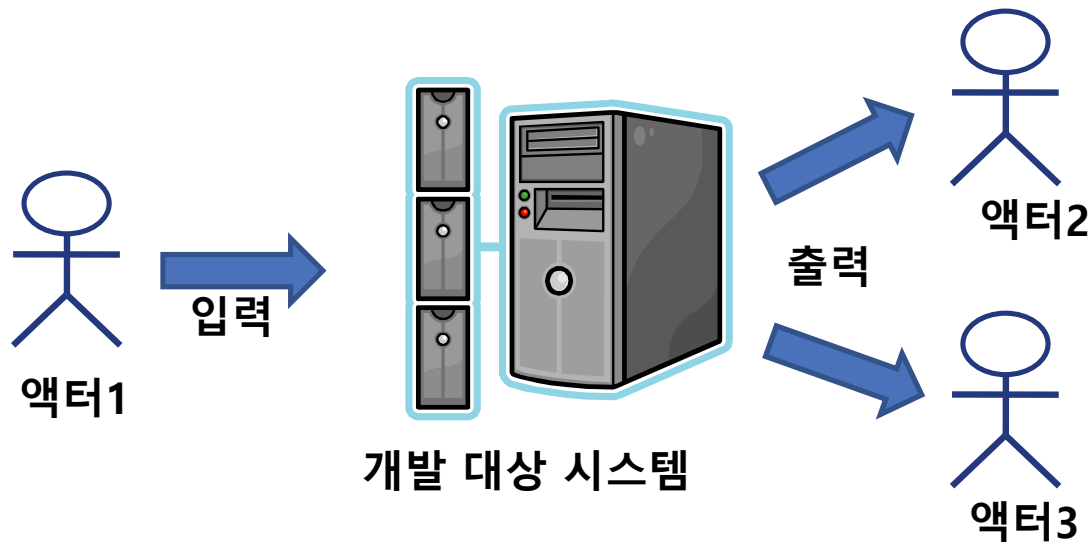
# 액터

---



# 액터

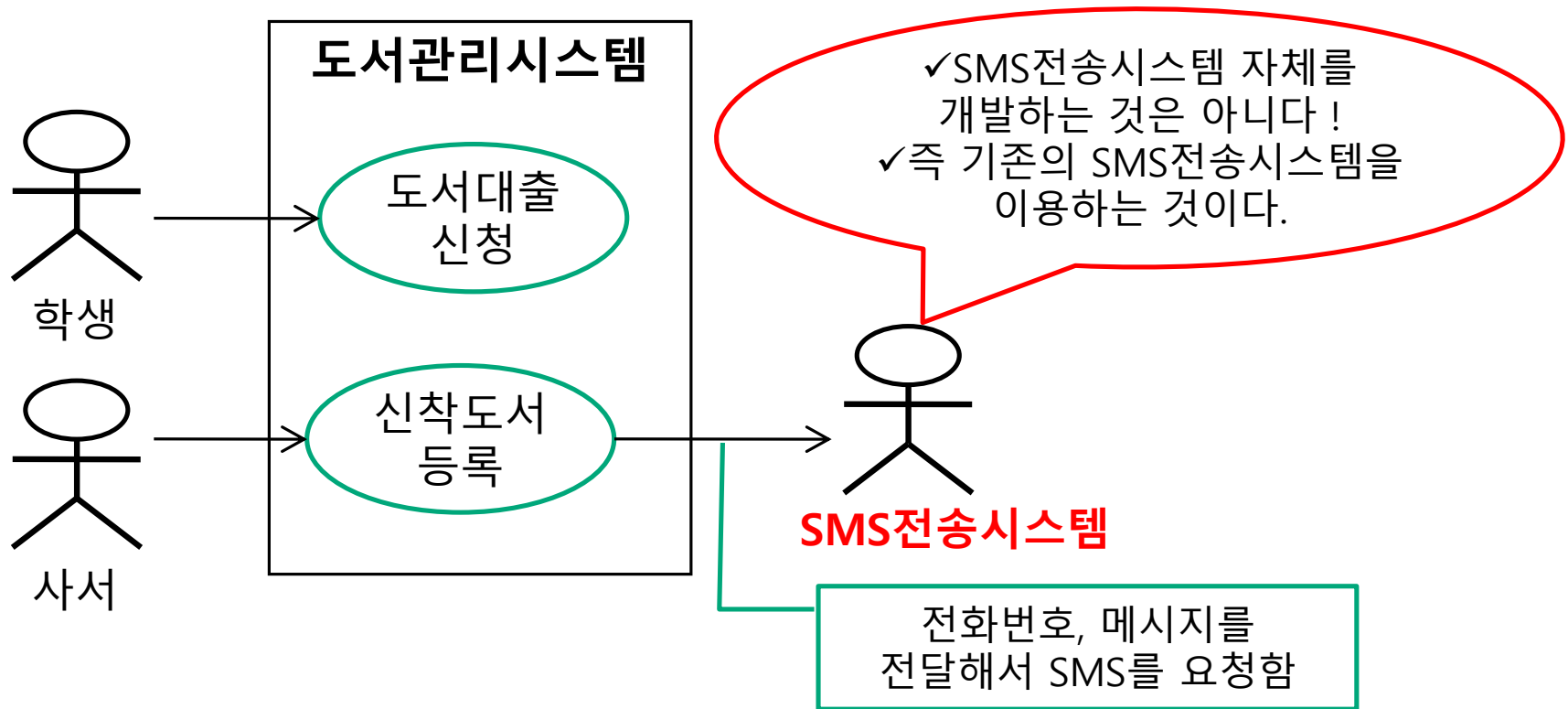
- 액터는 시스템과 상호작용(interaction)을 하는 시스템 외부의 존재이다.





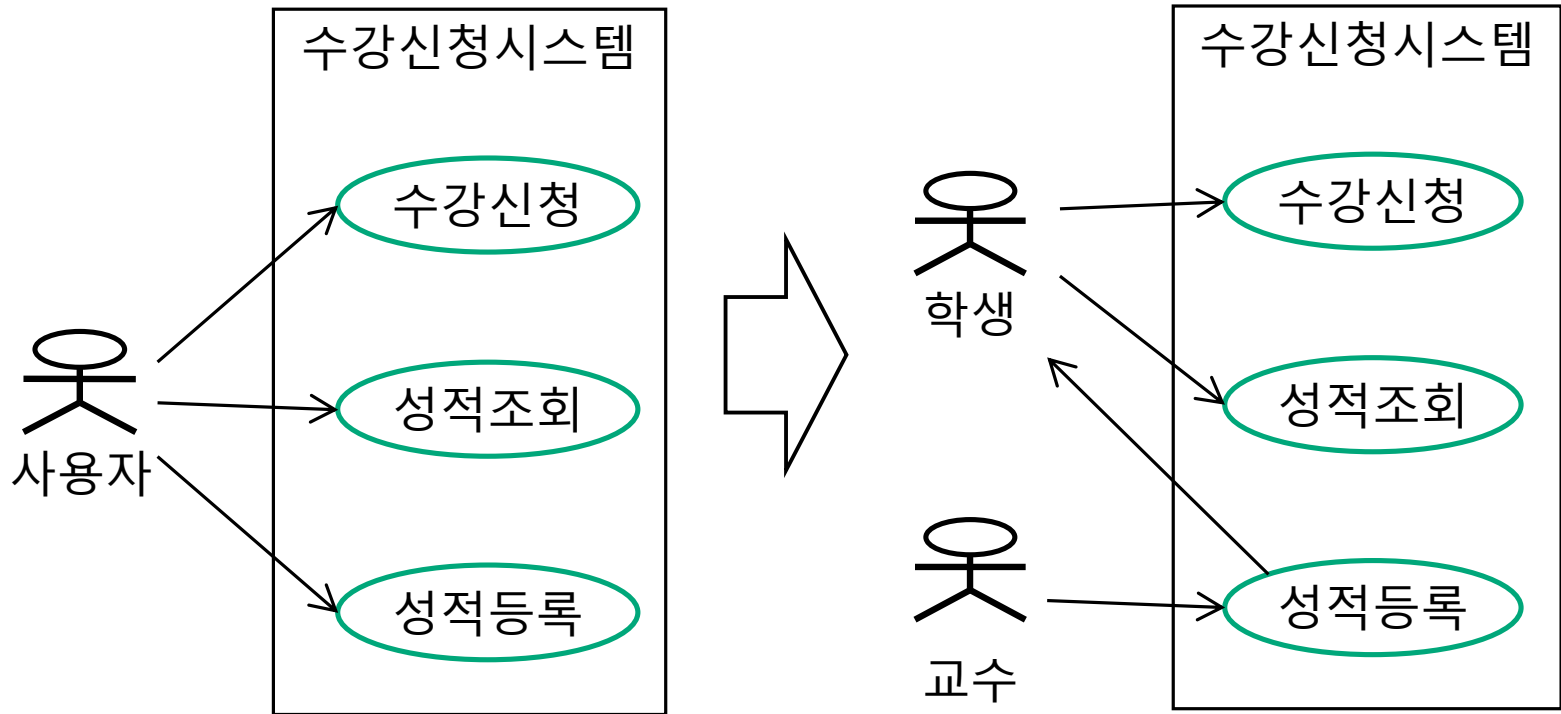
# 액터 - 시스템 외부의 대상 표현

- 액터는 시스템 외부의 존재이다.



# 액터 - 시스템 관점의 역할

- 액터는 시스템 관점에서 바라본 역할을 뜻해야 한다.



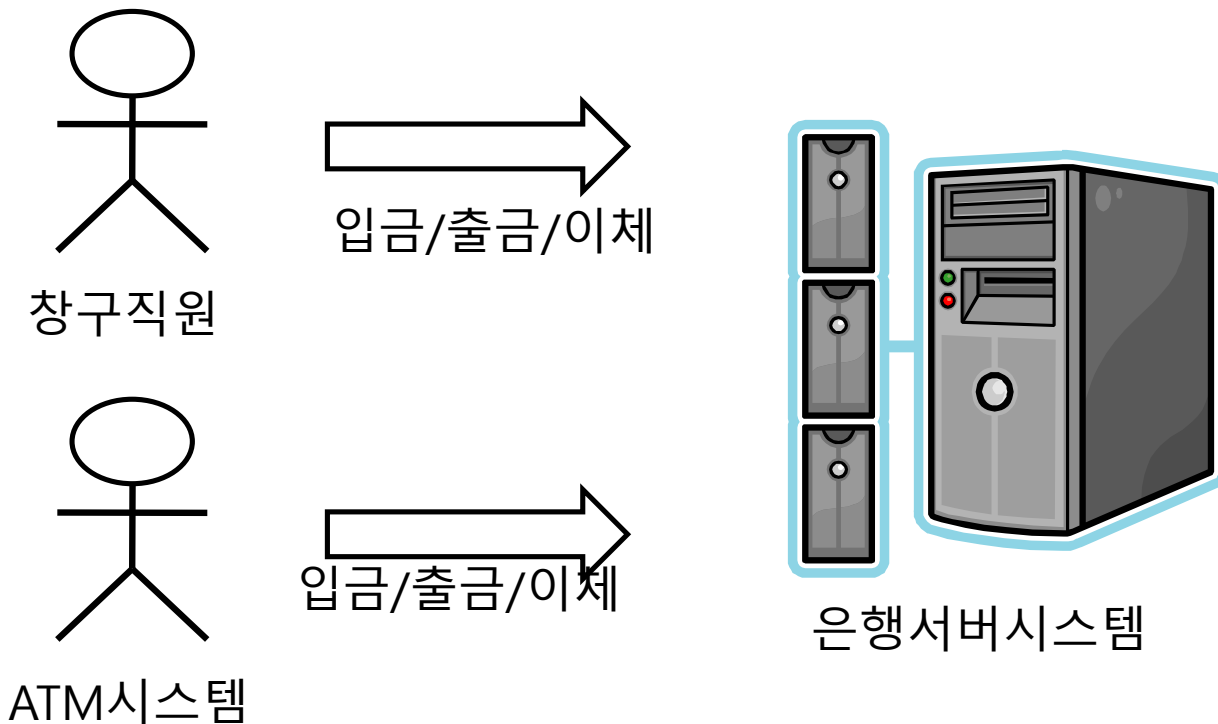
# 액터 – 개발 대상 시스템에 의해 결정

- 액터는 개발 대상이 되는 시스템에 따라서 달라질 수 있다.
- 예) ATM시스템의 액터



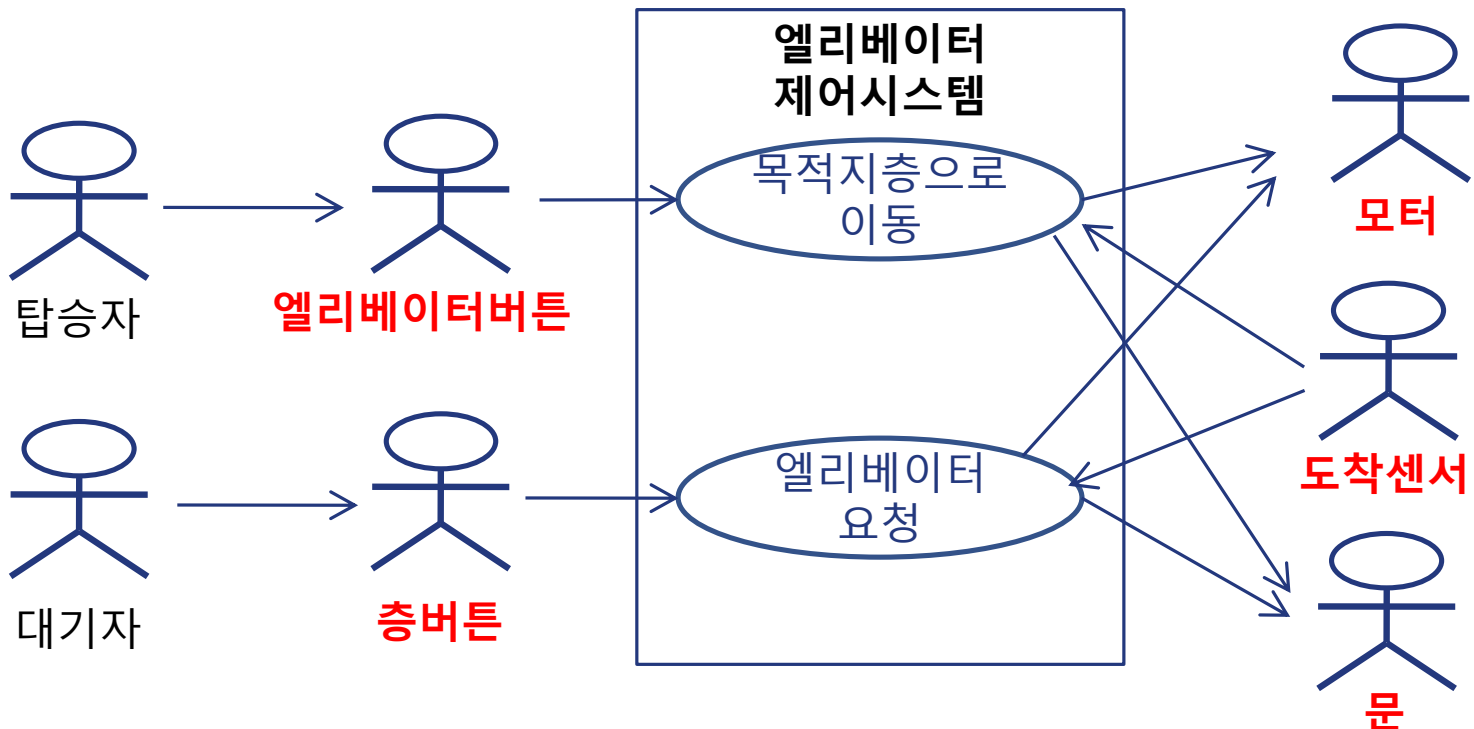
# 액터 – 개발 대상 시스템에 의해 결정

## ■ 예) 은행서버시스템의 액터



# 액터 - 장치 유형의 액터

- 장치 유형의 액터의 표현 여부는 표준 플랫폼에 의한 지원에 따라 달라진다.



# 액터(Actor) - 요약

---

- 액터는 시스템과 상호작용(interaction)을 하는 시스템 외부의 존재이다.
- 액터는 시스템과 직접적인 상호작용을 해야 한다.
- 액터는 시스템 관점에서 바라본 사용자의 역할을 뜻해야 한다.
- 액터는 개발 대상이 되는 시스템에 따라서 달라질 수 있다.
- 장치 유형의 액터의 표현 여부는 표준 플랫폼에 의한 지원에 따라 달라진다.



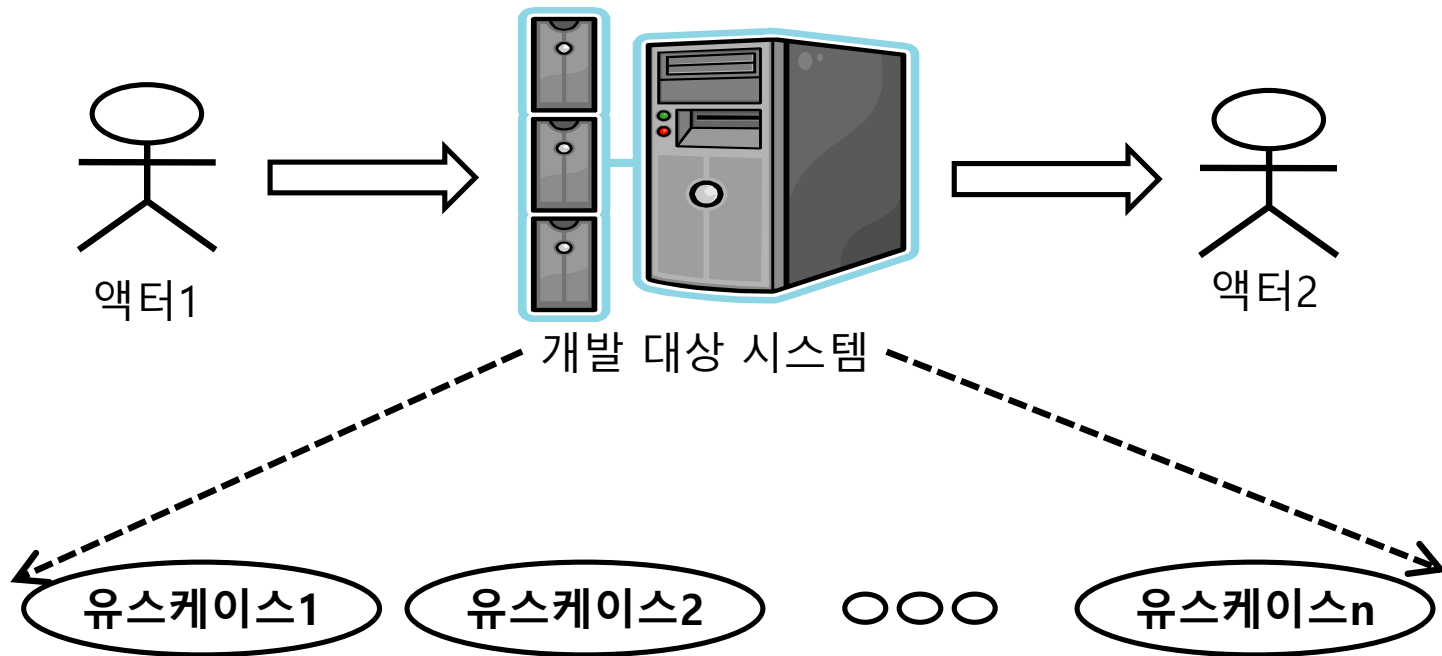
# Use case

---



# 유스케이스 – 시스템의 기능 단위

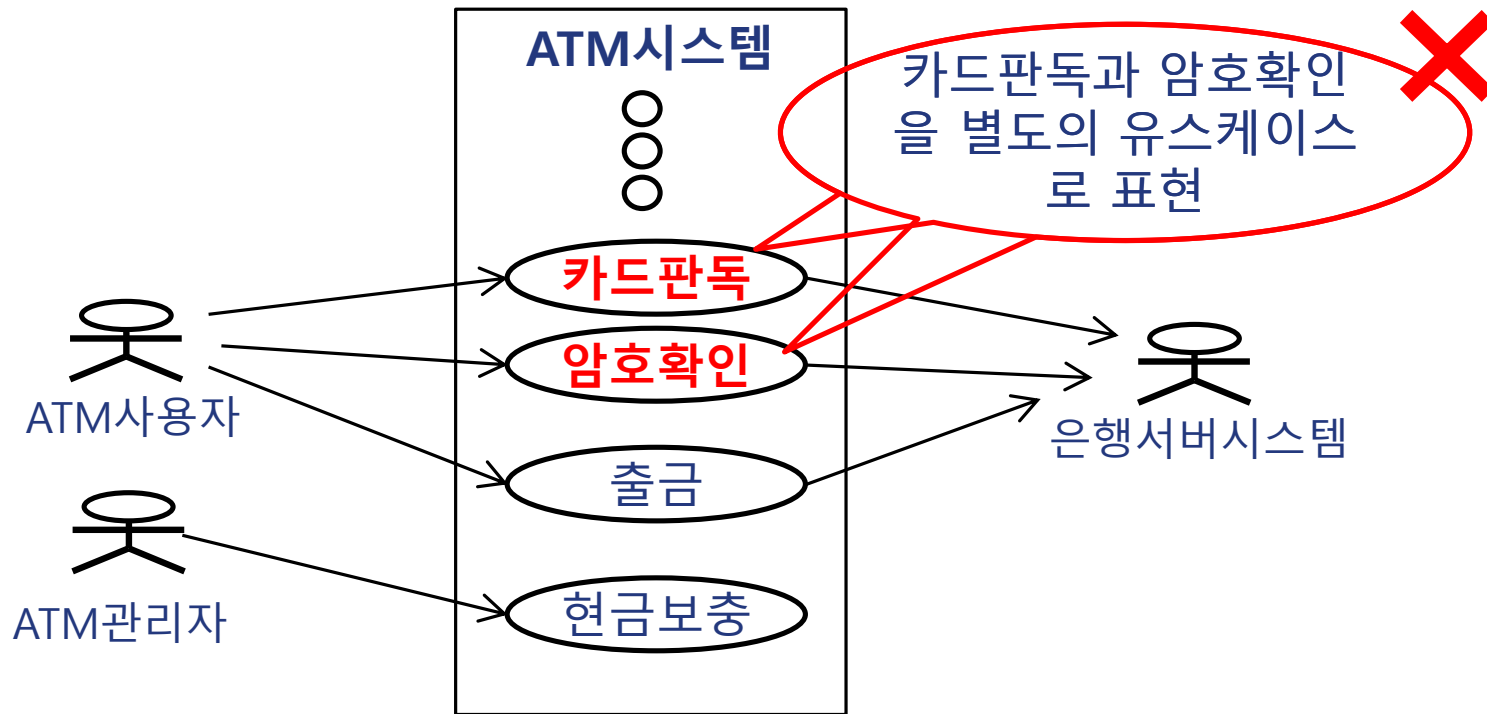
- 시스템이 제공하는 개별적인 기능을 뜻한다





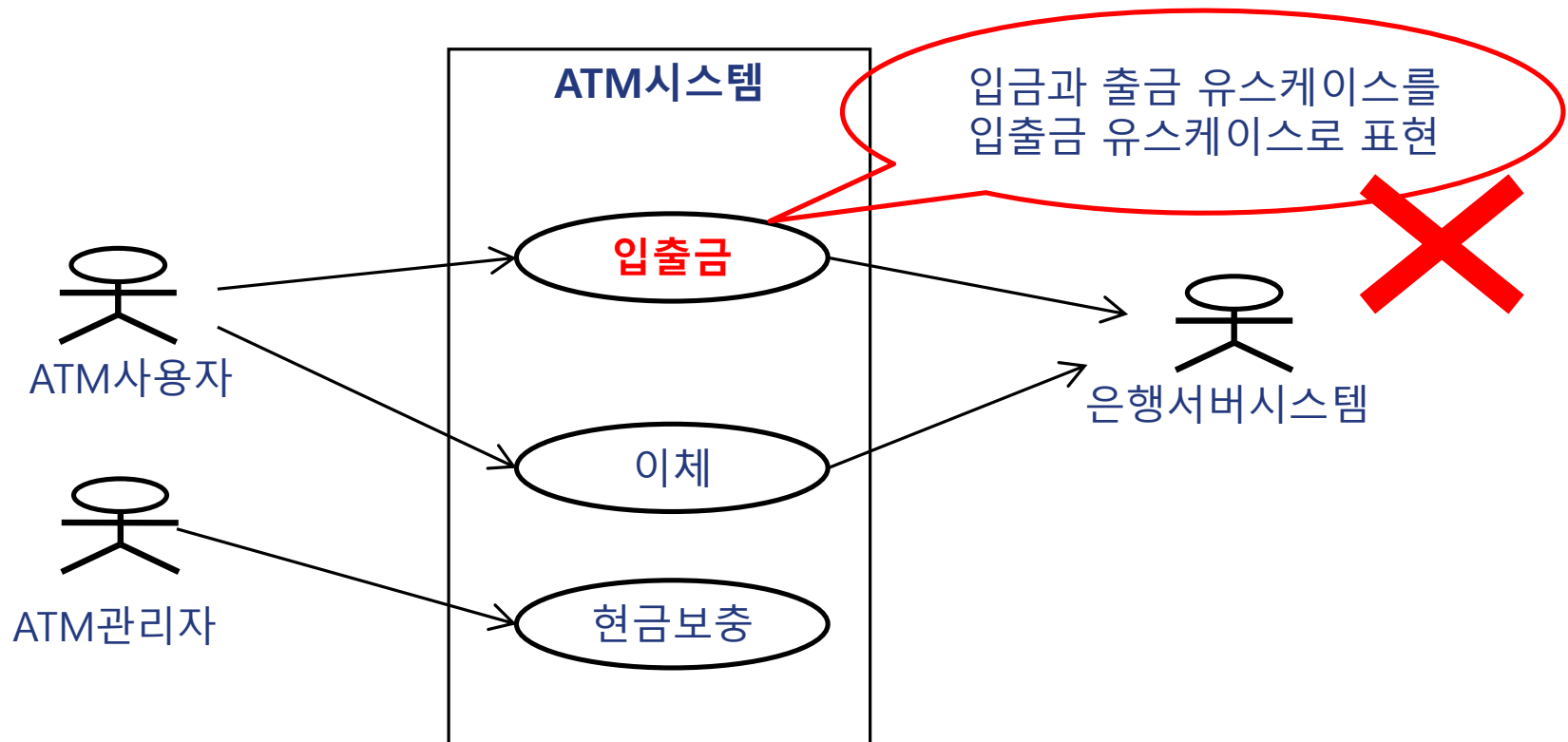
# 유스케이스 - 사용자의 인식 기능 단위

- 사용자가 인지할 수 있는 하나의 기능 단위이다.



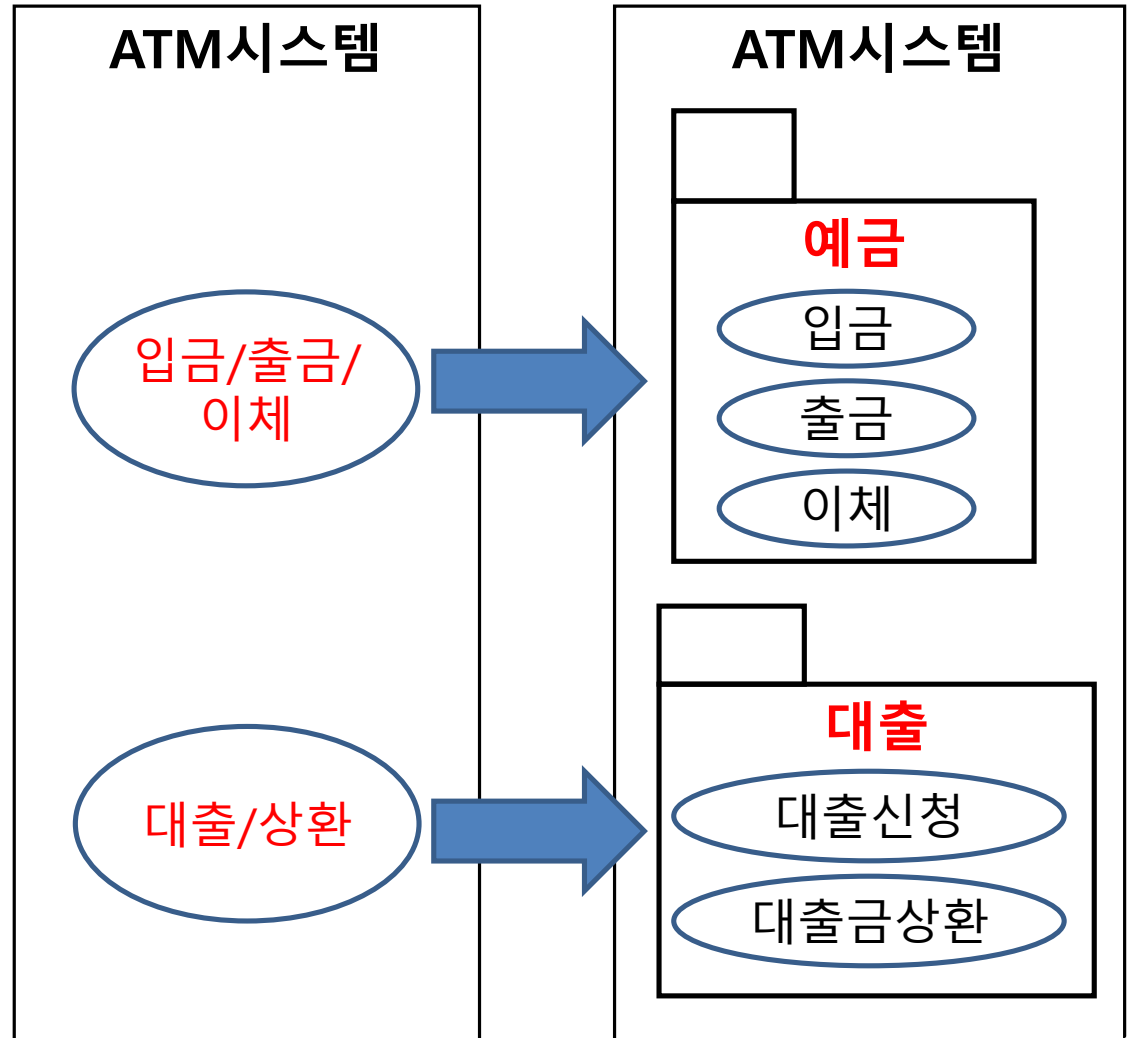
# 유스케이스 - 구체성

- 구체적(concrete)이어야 한다. 실제로 현실에서 발생하는 기능이어야 한다.



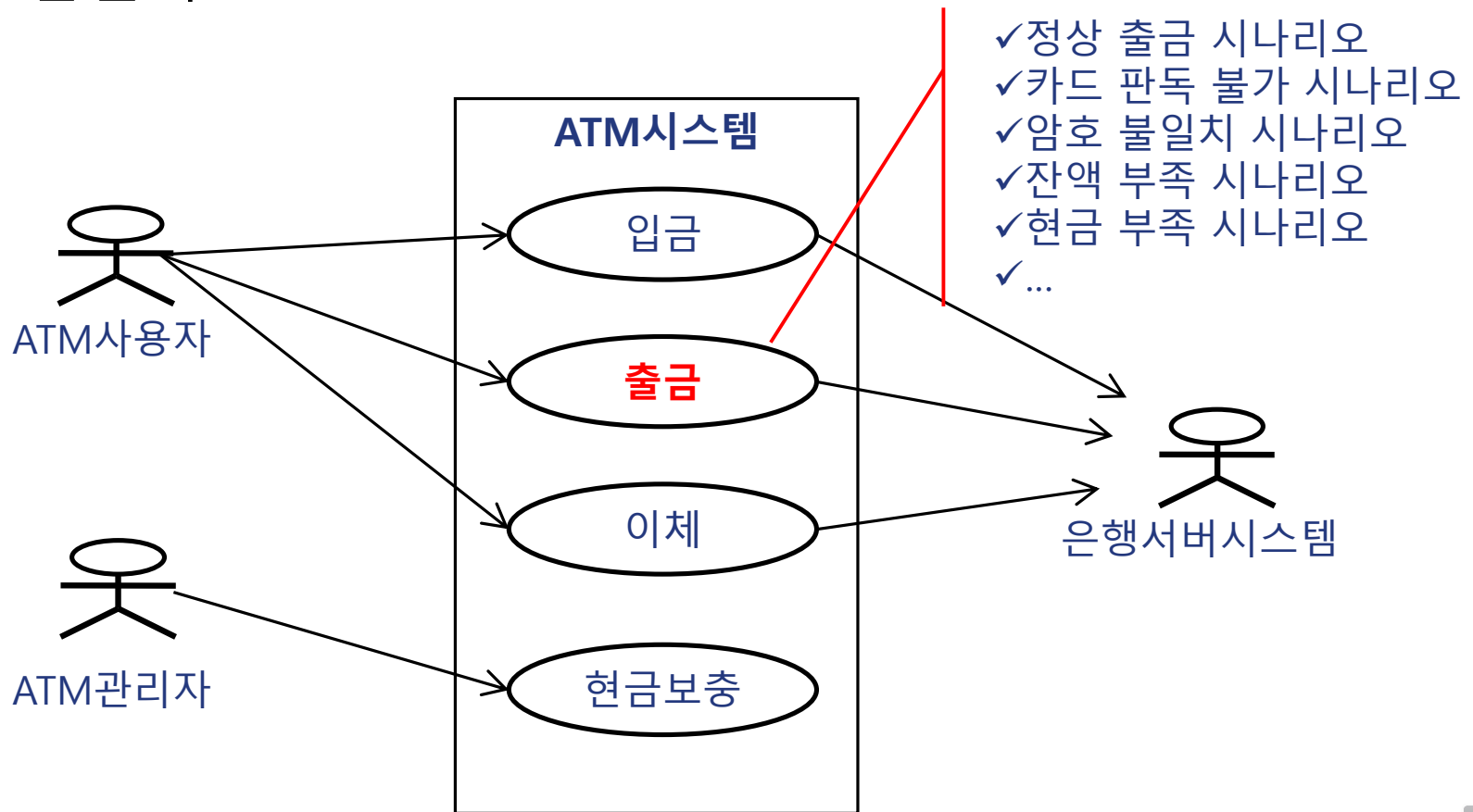
# 유스케이스 - 구체성

- 유스케이스 패키지의 사용



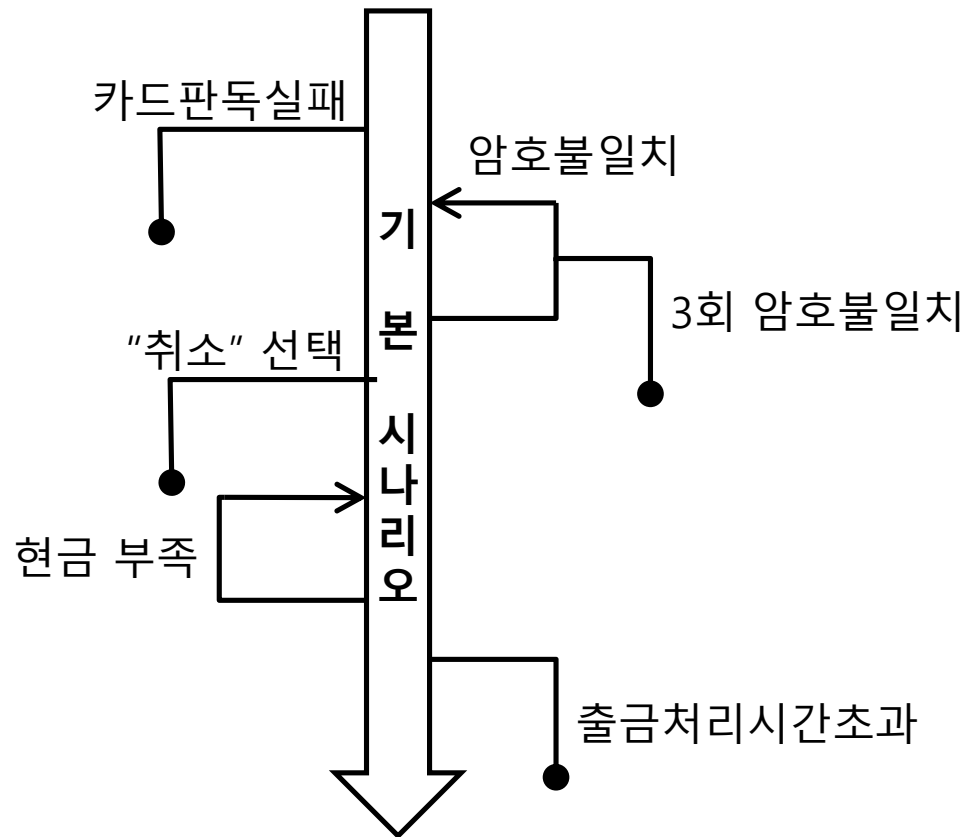
# 유스케이스 - 다수 시나리오 포함

- 하나의 유스케이스는 다양한 세부 상황(시나리오)을 포함한다.



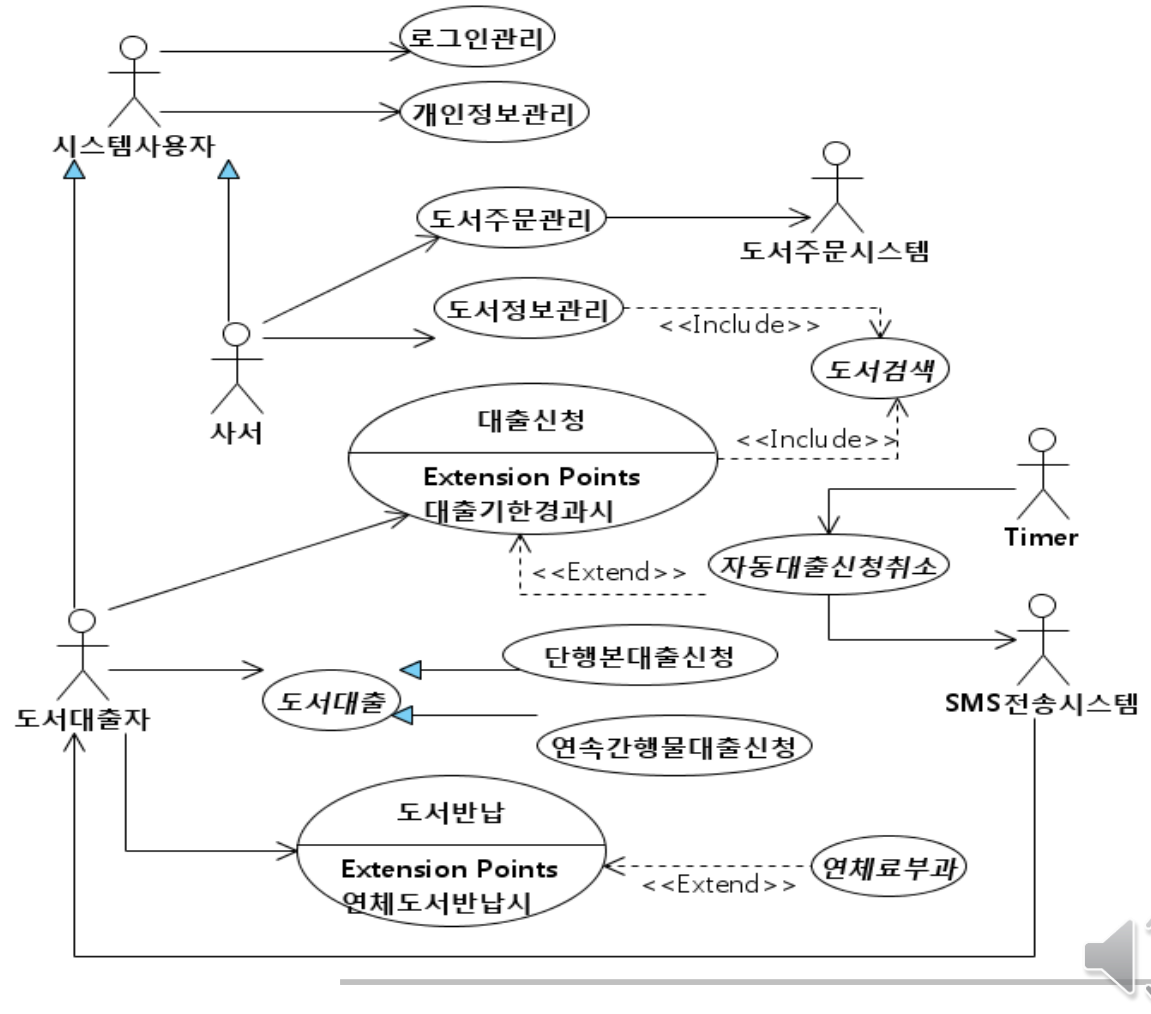
# 유스케이스 – 다수 시나리오 포함

- 하나의 유스케이스는 여러 시나리오를 포함한다.



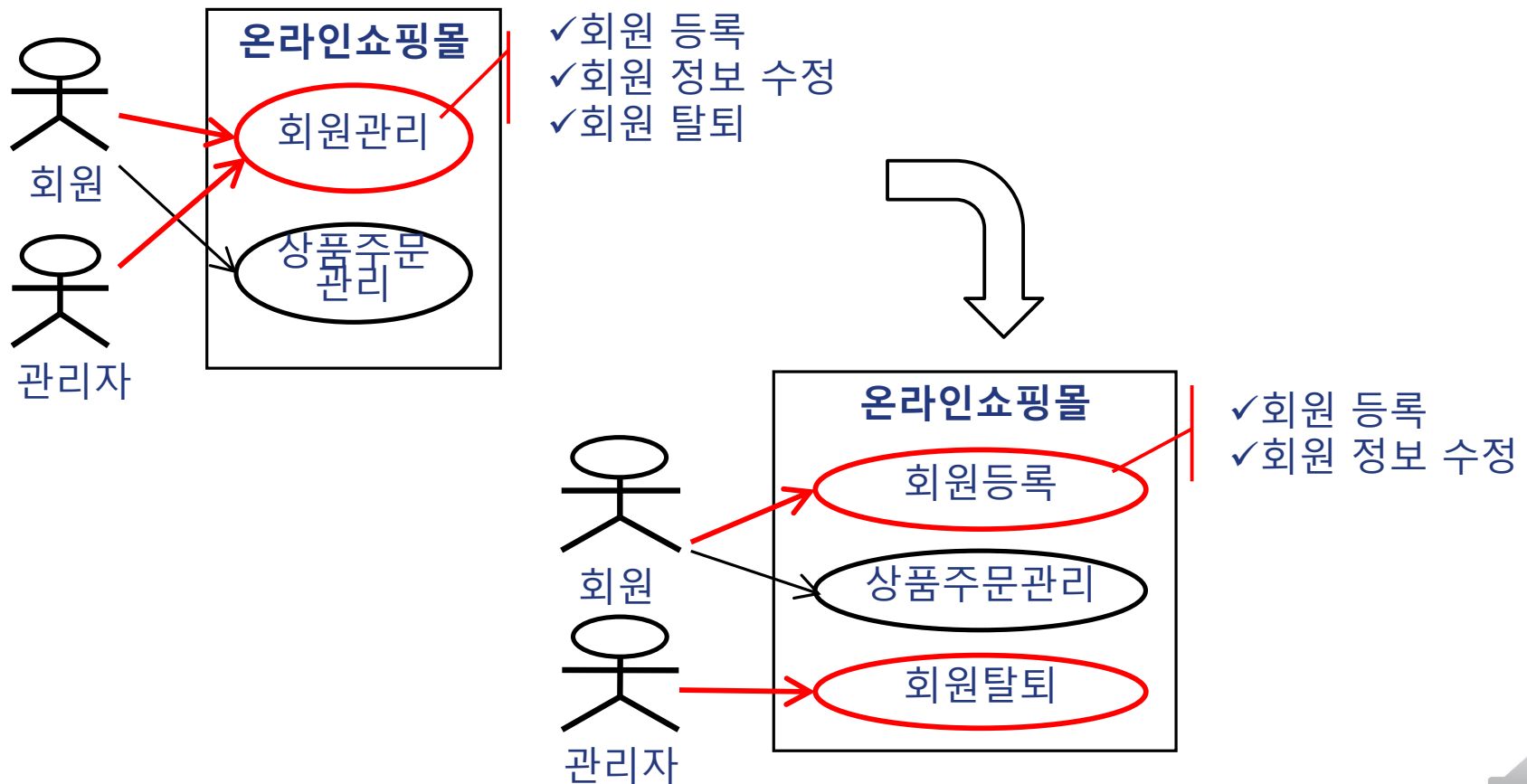
# 유스케이스 – 활성화 액터의 존재

- 활성화(triggering) 상호작용을 하는 액터가 있어야 한다.  
➔ Primary actor



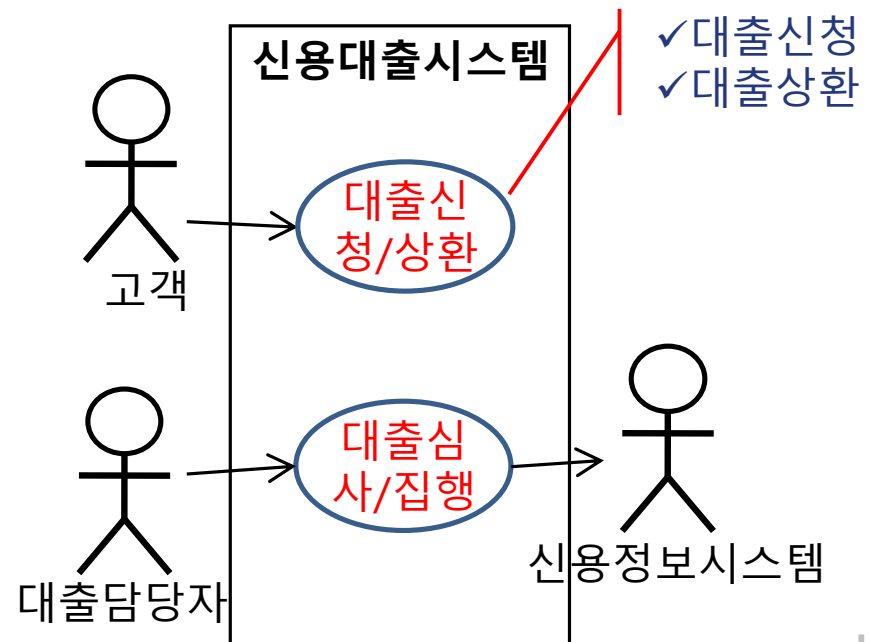
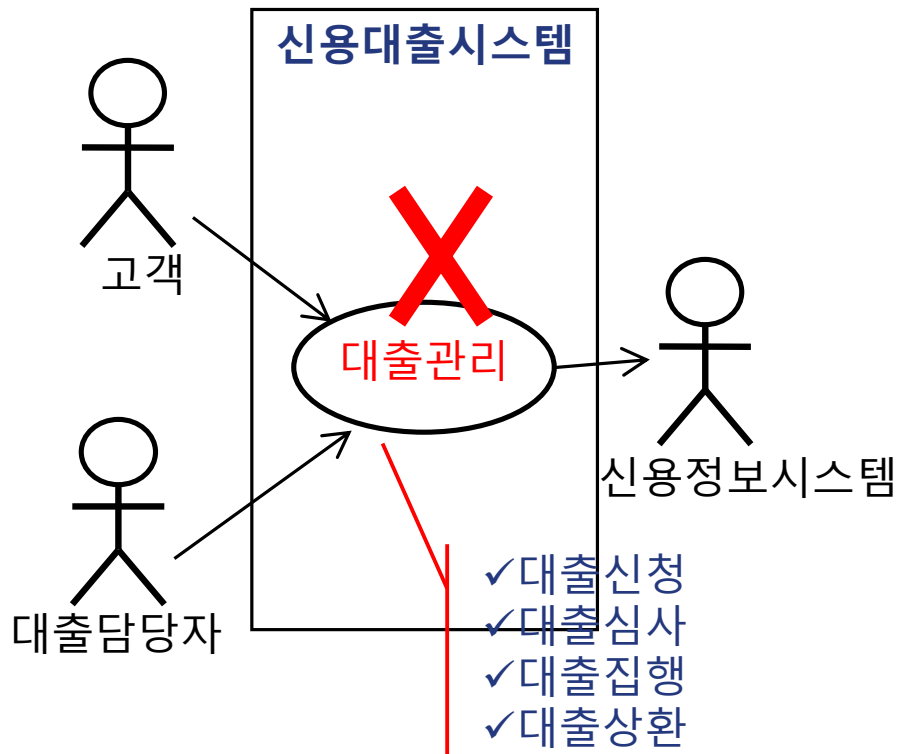
# 유스케이스 - 동일 기능을 제공

- 모든 활성화 액터에게 동일한 기능을 제공해야 한다.



# 유스케이스 - 트랜잭션

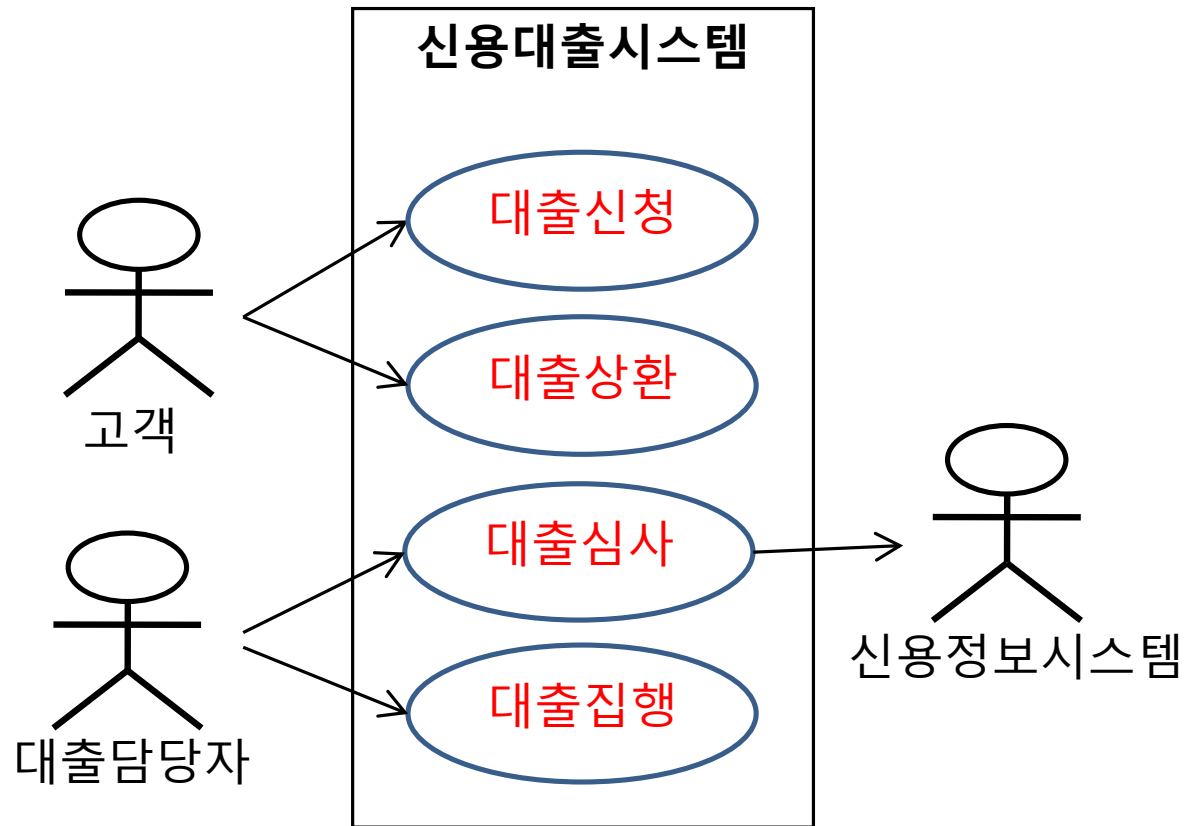
- 유스케이스는 트랜잭션 성격을 가져야 한다.





# 유스케이스 - 트랙잭션

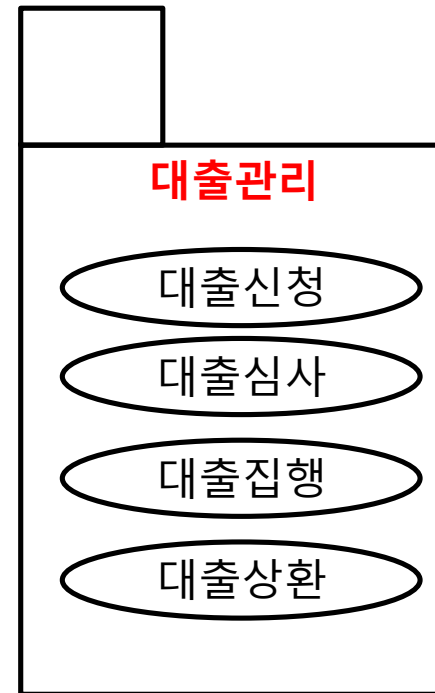
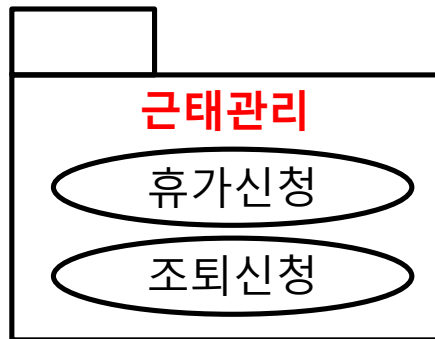
## ■ 올바른 예



# 유스케이스 – CRUD 유스케이스

---

- 데이터(정보)에 대한 CRUD는 하나의 유스케이스로 표현한다.



# 유스케이스 - 명명 방법

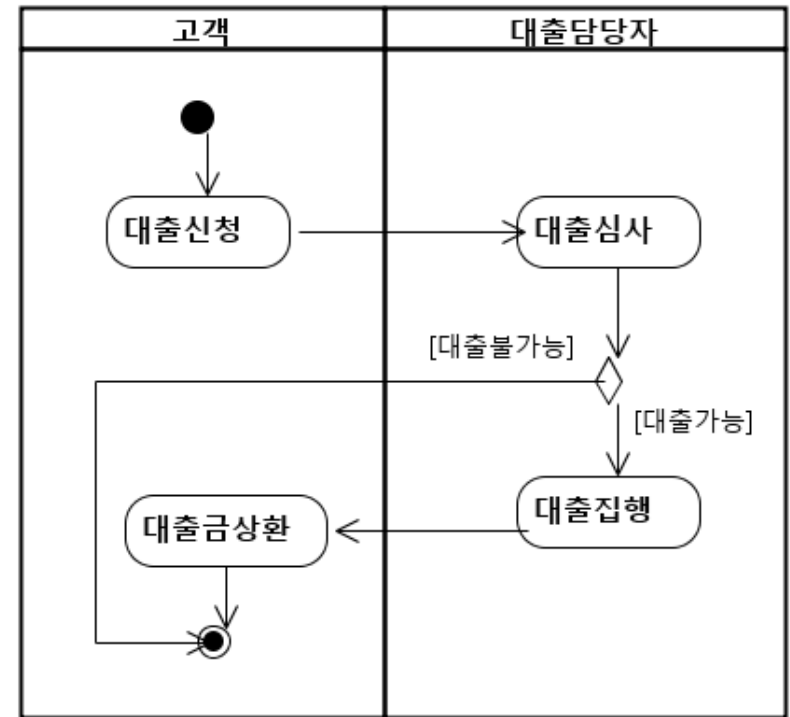
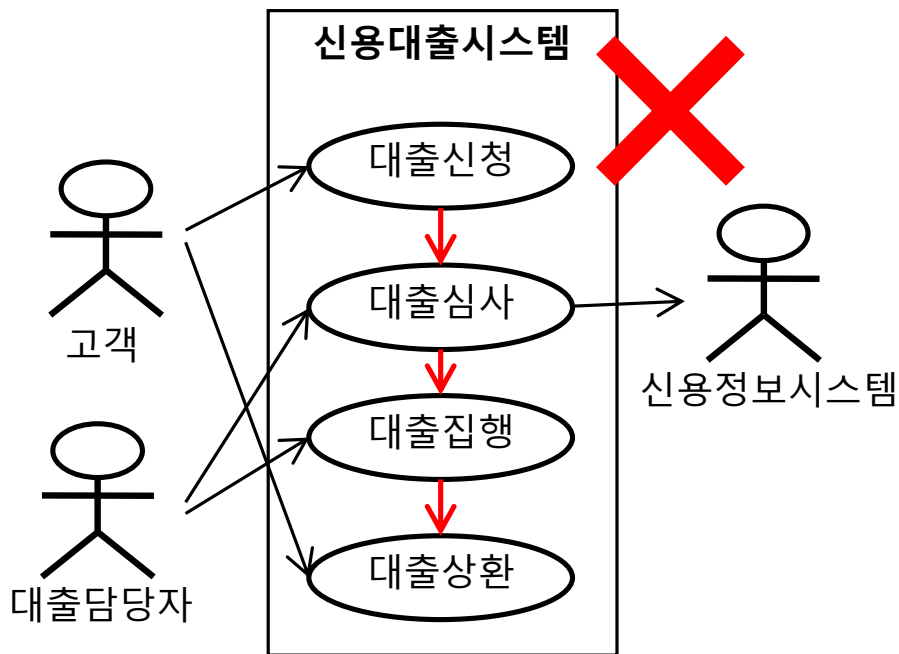
---

- 유스케이스를 통하여 제공되는 시스템의 기능을 표현하는 구체적이고 명확한 동사구 형태

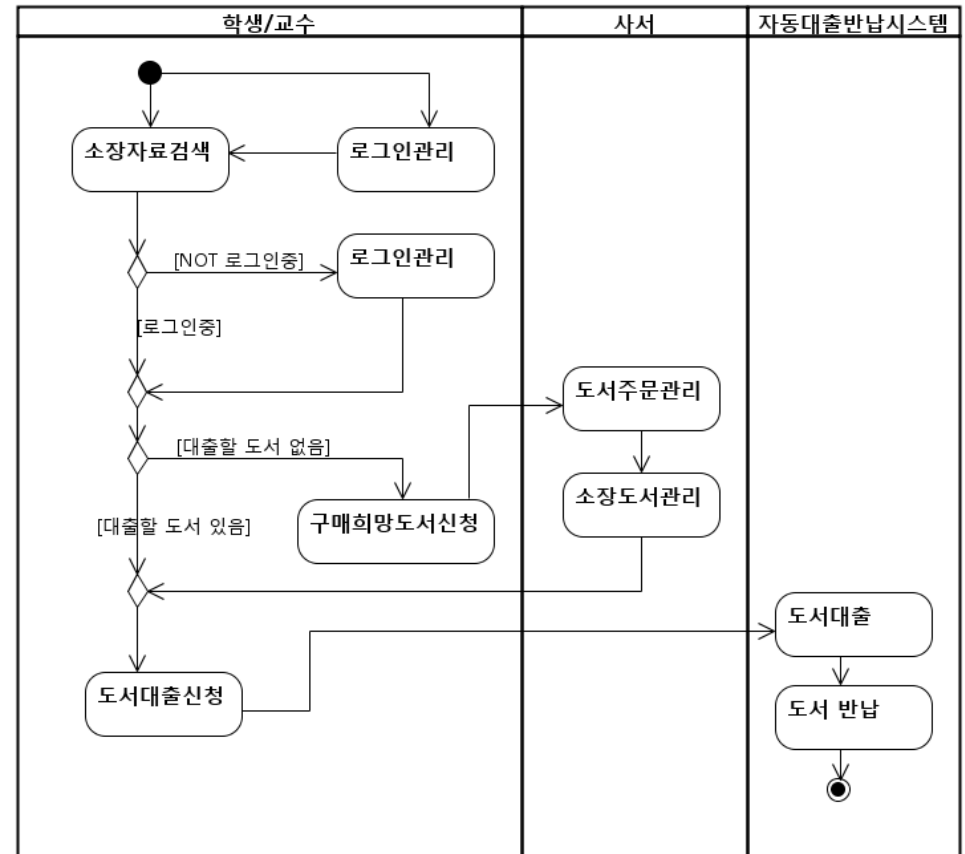
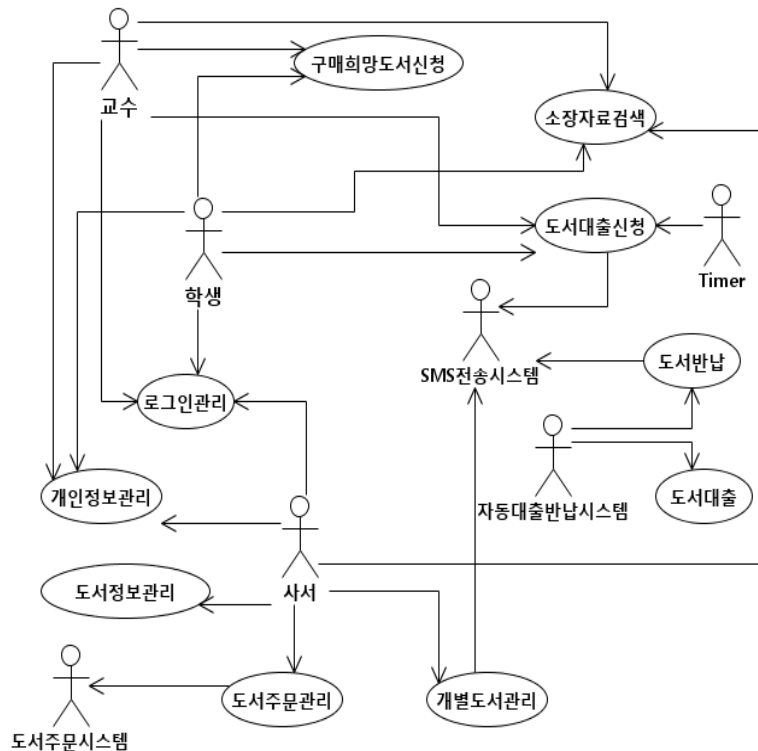


# 유스케이스 - 선/후행 관계의 표현

- 유스케이스 간의 선/후행 관계는 액티비티 다이어그램을 이용해서 표현할 수 있다.



# 유스케이스 - 선/후행 관계의 표현



# 유스케이스(Use case) - 요약

---

- 유스케이스는 사용자가 인지할 수 있는 하나의 기능 단위이다.
- 유스케이스는 구체적(concrete)이어야 한다.
- 하나의 유스케이스로 다양한 세부 상황을 포함한다.
- 반드시 한 개 이상의 활성화 상호작용을 하는 액터가 있어야 한다.
- 유스케이스는 모든 활성화 액터에게 동일한 기능을 제공해야 한다.
- 유스케이스는 트랜잭션 성격을 가져야 한다.
- 데이터(정보)에 대한 CRUD는 하나의 유스케이스로 표현한다.



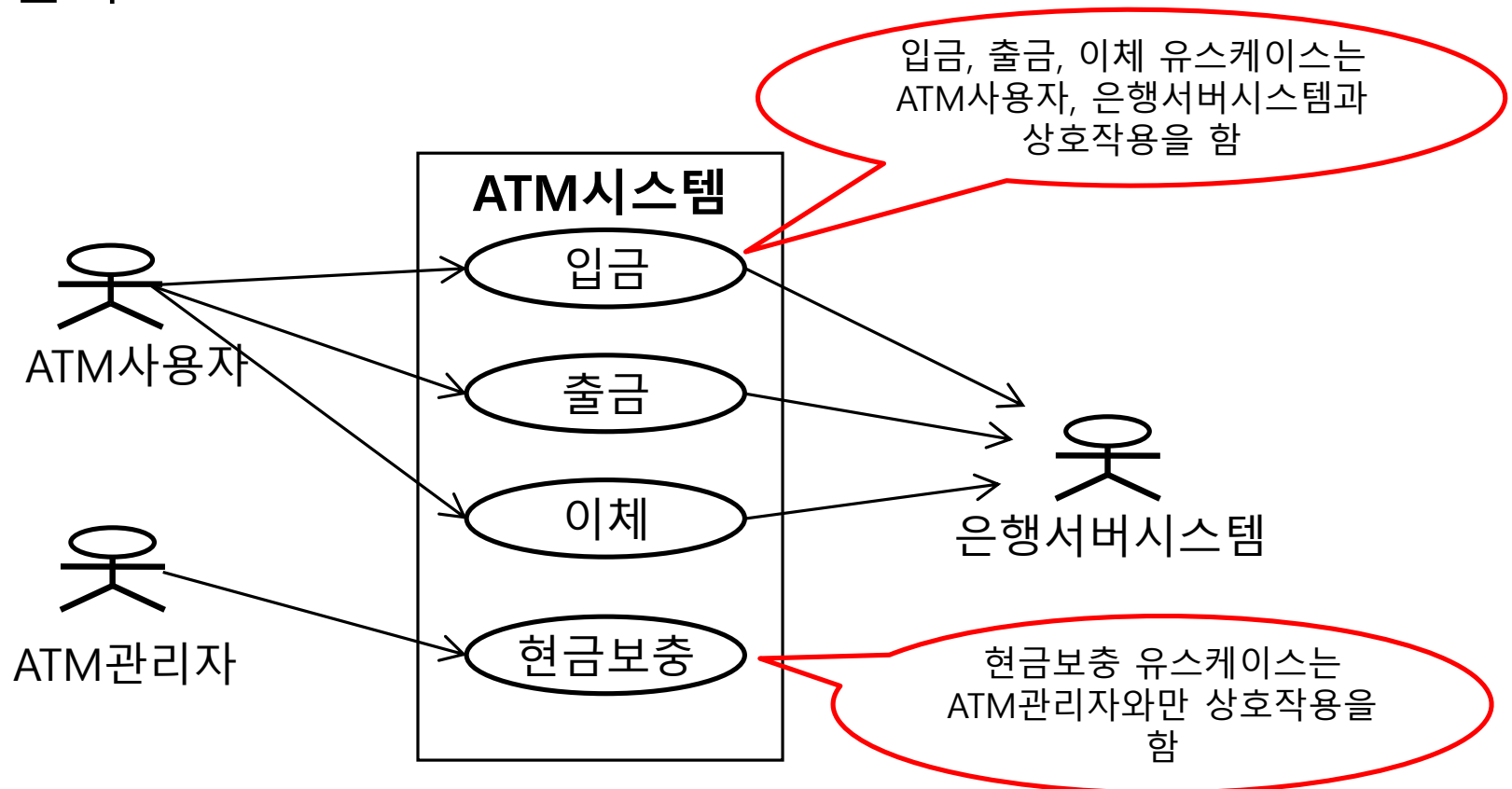
# 액터와 유스케이스간의 관계

---



# 상호작용의 표현

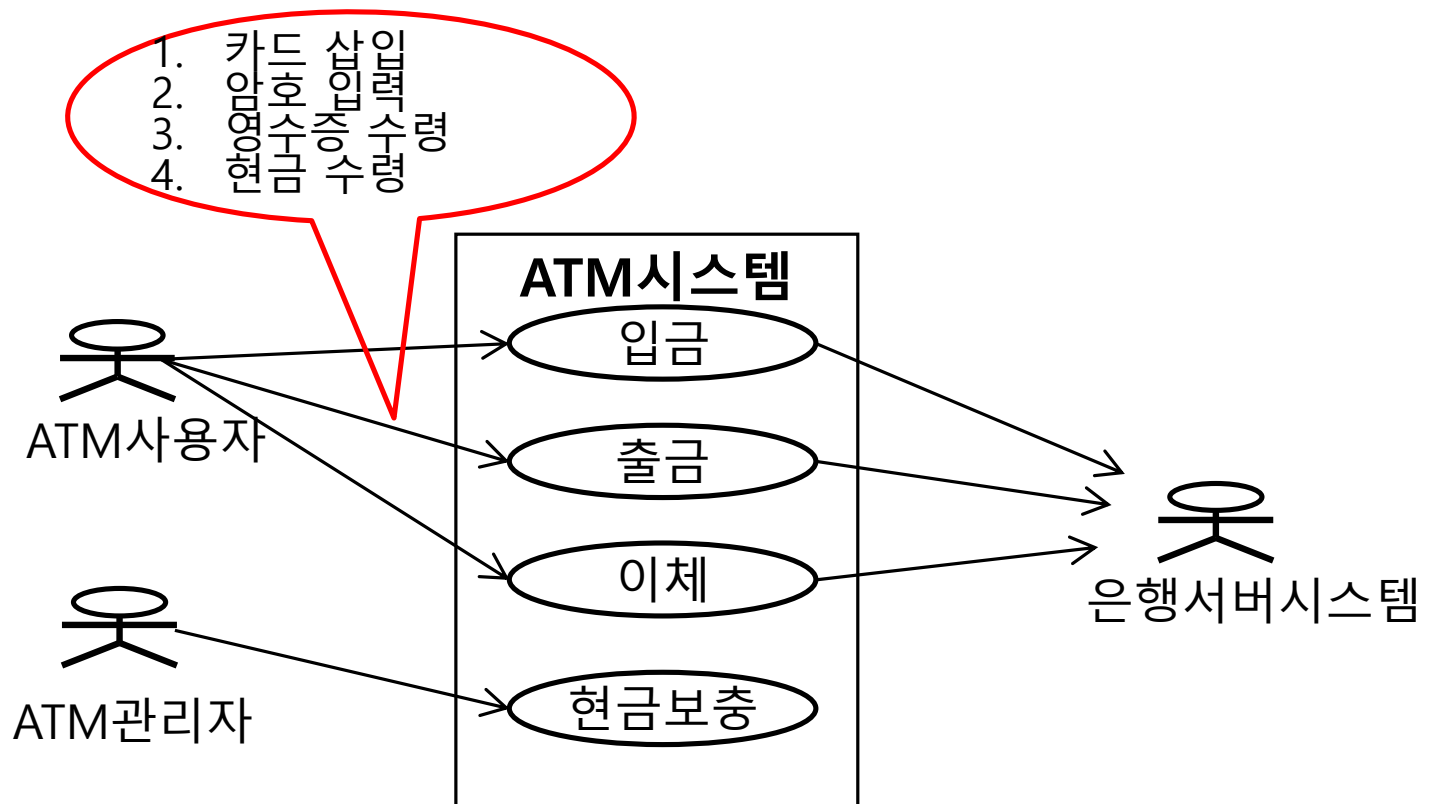
- 액터와 유스케이스 간의 연관 관계는 둘 간의 상호작용을 뜻한다.



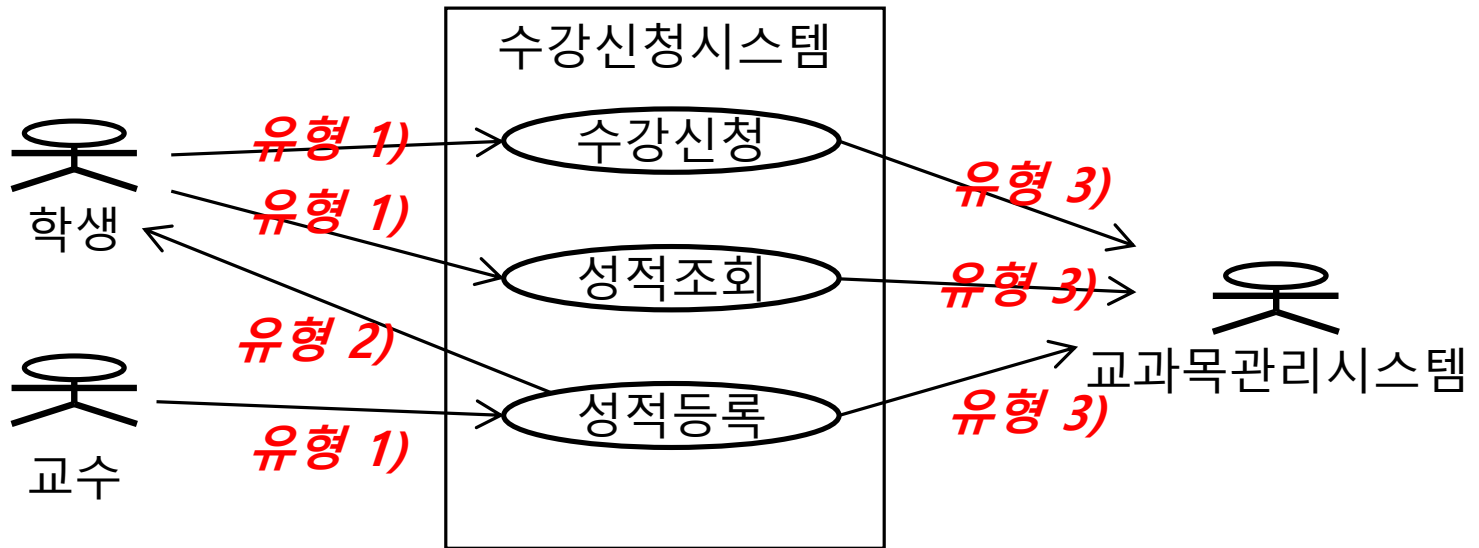


# 다양한 상호작용의 포함

- 하나의 연관관계는 시스템과 액터간의 다양한 상호작용을 표현한다.



# 액터와 유스케이스 간의 관계 - 유형

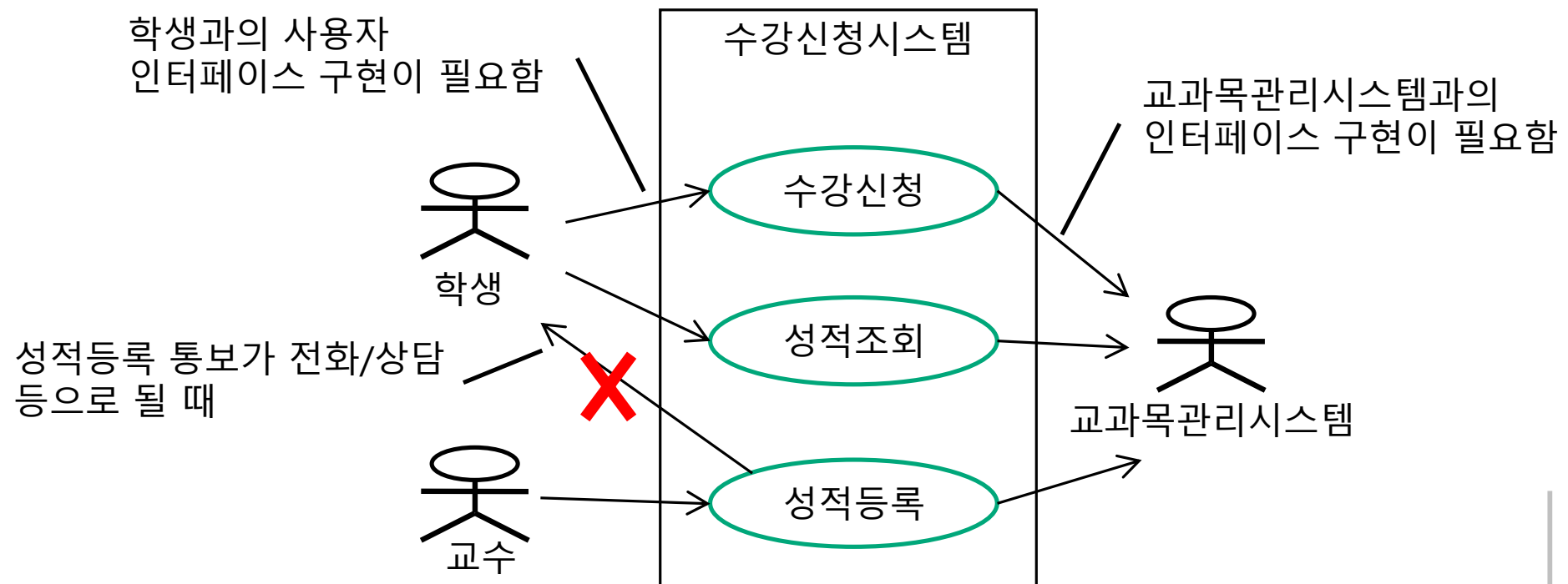


유형	설명	연관 관계의 방향
1) 활성화	액터가 유스케이스를 활성화 시킴	
2) 수행결과 통보	유스케이스 결과가 액터에게 통보됨	
3) 외부서비스 요청	외부 시스템의 서비스 실행을 요청함	



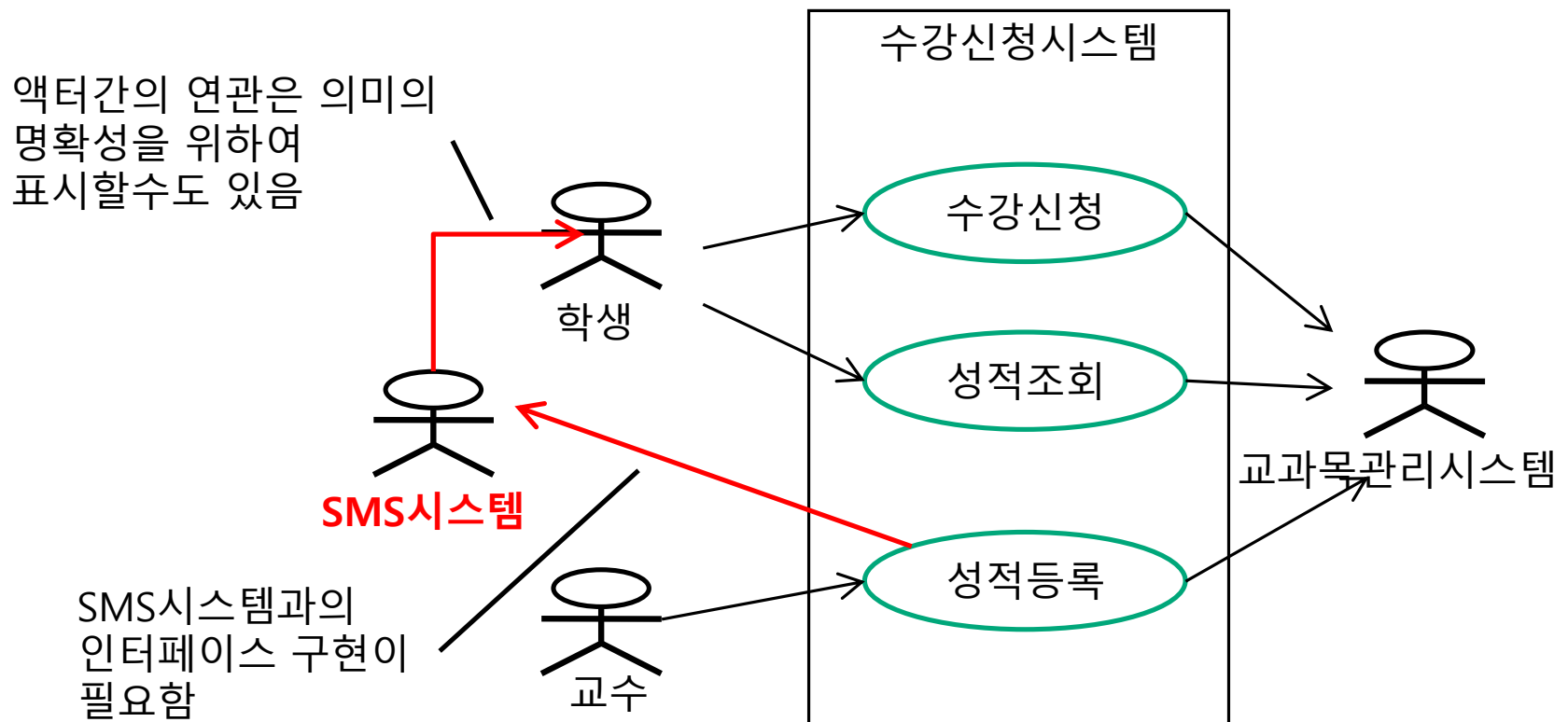
# 시스템 인터페이스 기능의 표현

- 연관관계는 반드시 시스템이 제공하는 기능이어야 한다.



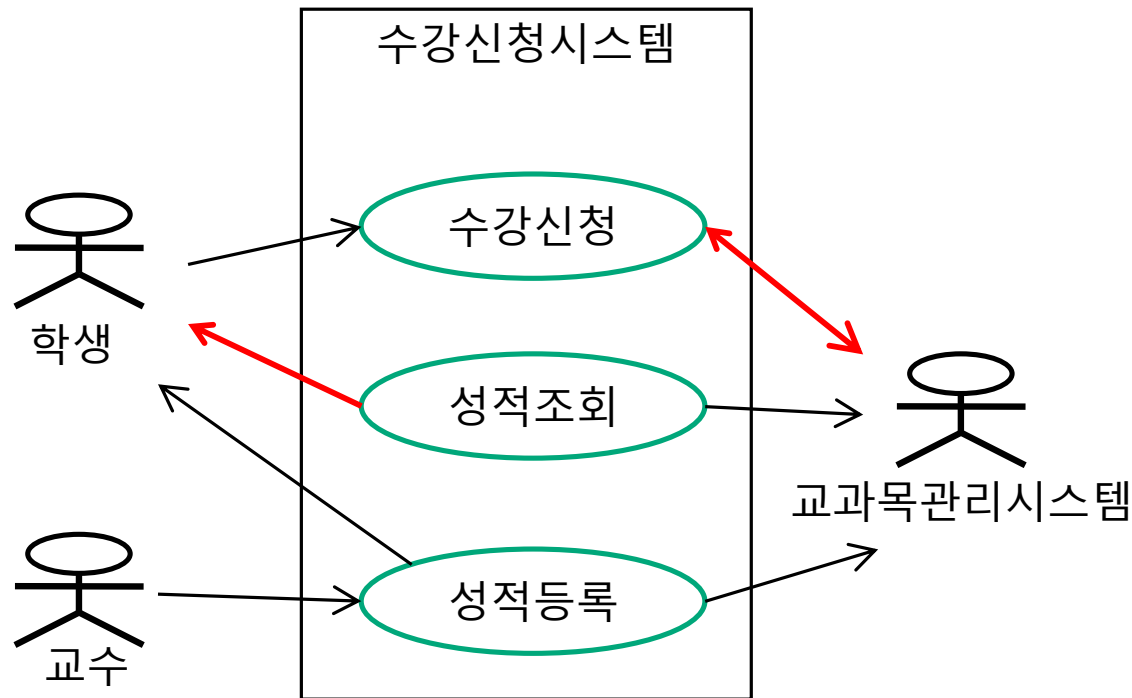
# 시스템 인터페이스 기능의 표현

- 예) 성적등록을 SMS시스템을 이용해서 자동으로 할 때



# 연관관계의 방향

- 연관관계의 방향은 데이터의 흐름이 아니라 제어 흐름을 뜻해야 한다.



## 액터와 유스케이스 간의 관계 - 요약

---

- 액터와 유스케이스 간의 연관 관계는 둘 간의 상호작용을 뜻한다.
- 액터는 유스케이스와의 연관관계를 통하여 시스템과 다양한 상호작용을 한다.
- 연관관계는 반드시 시스템이 제공하는 기능이어야 한다.
- 연관관계의 방향은 제어 흐름을 뜻해야 한다.



# Q&A

---

