



CSE053 Software Engineering

Lecture: Use case diagram (II)

Software engineering laboratory
Yeungnam university



Learning objectives

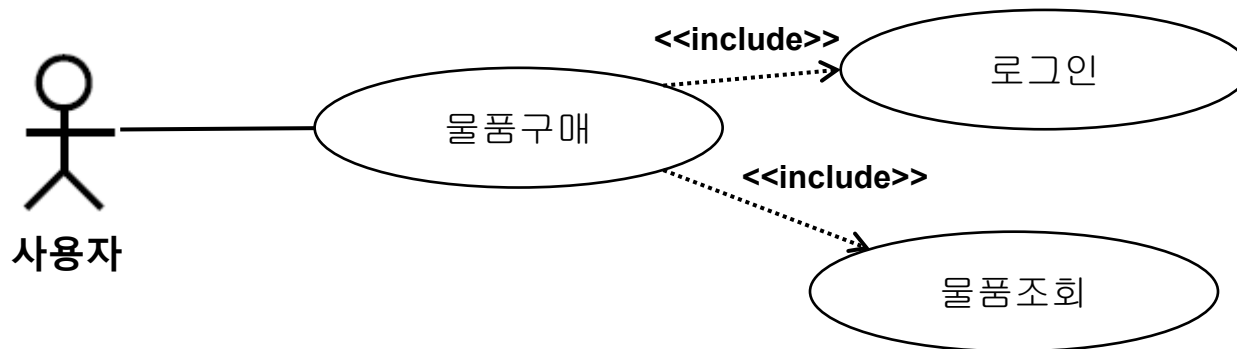
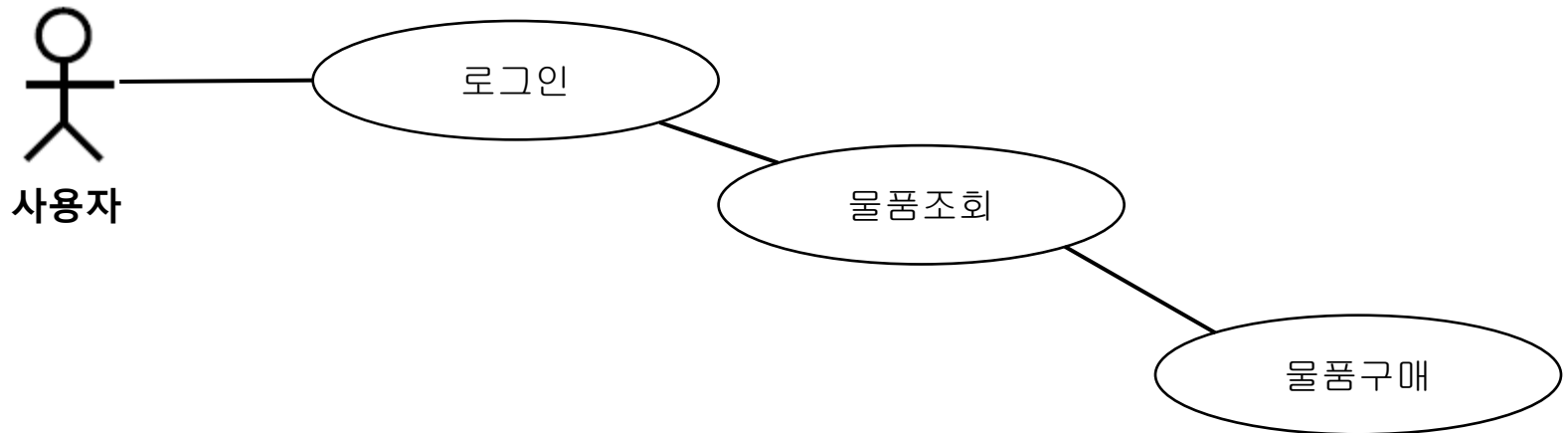
- Use case diagram 설계시 유의사항을 이해한다.
- Use case description을 이해한다.
- Use case description을 실습한다.



Use case diagram의 올바른 이해

■ Use case diagram은 흐름도가 아니다 !

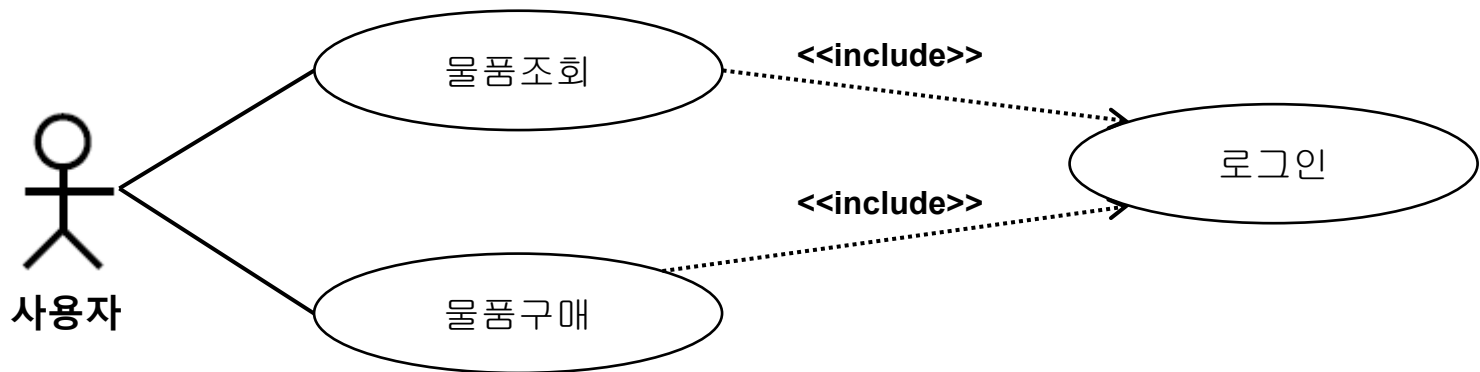
- 잘못된 예



Use case diagram의 올바른 이해

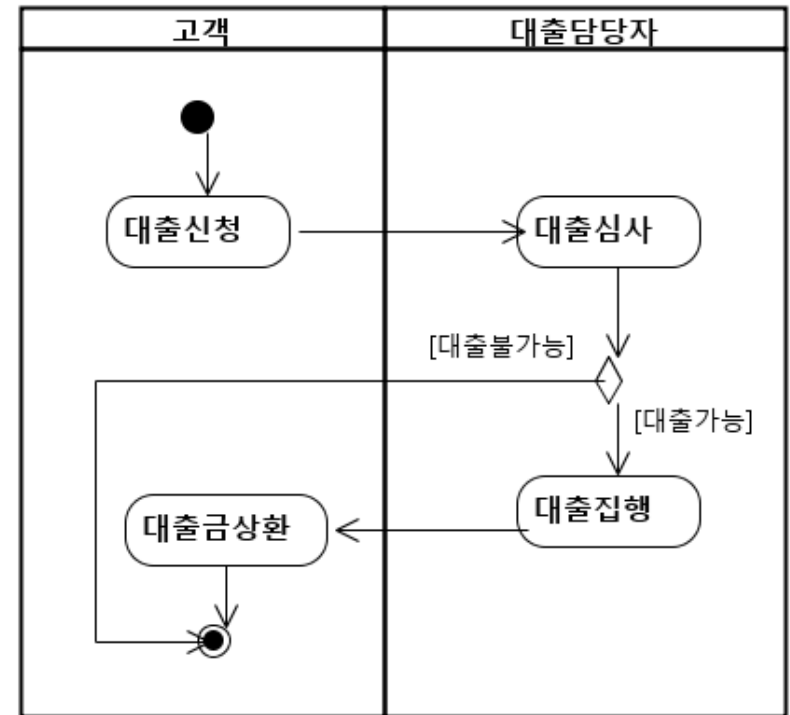
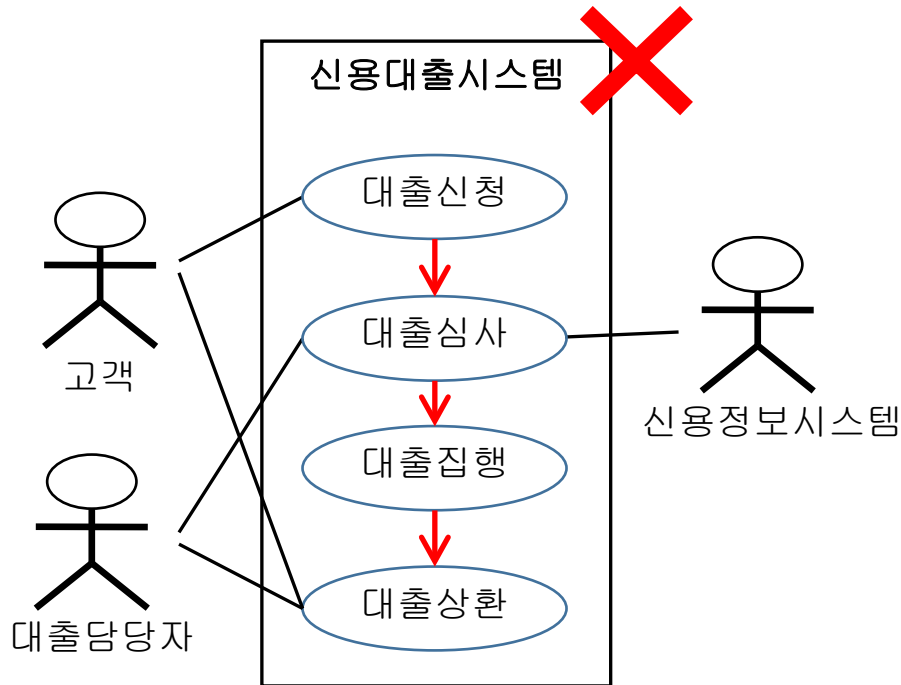
- Use case diagram은 흐름도가 아니다 !

- 올바른 예



Use case diagram의 올바른 이해

- Use case 간의 선/후행 관계는 Activity diagram을 이용해서 표현



Use case diagram 모델링 절차

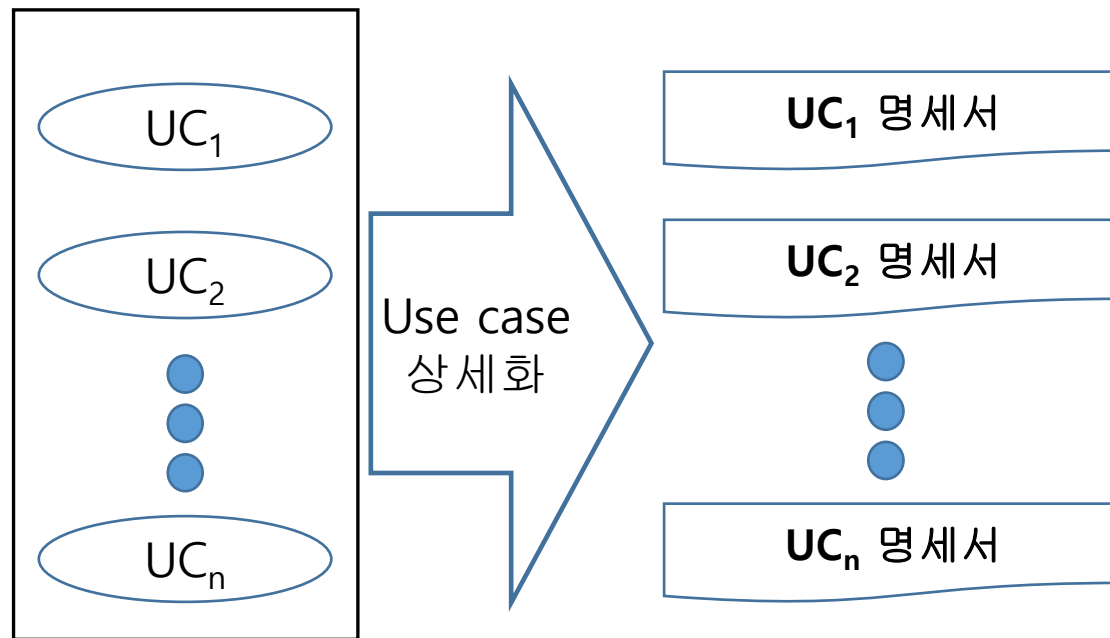
■ Use case diagram을 만드는 단계

- 1단계 : 시스템 상황을 확인 (Requirement 분석)
- 2단계 : Actor 식별
 - 행위자와 그들의 책임을 확인
- 3단계 : Use case 식별
 - 특정한 목적의 관점에서 볼 때 쓰임새와 시스템의 특성을 확인
- 4단계 : Association 식별 및 Use case diagram 작성
 - 행위자와 Use case에서 정제할 부분이 있는지 평가
 - Use case에서 <<include>> 의존성이 있는지 평가
 - Use case에서 <<extend>> 의존성이 있는지 평가
 - 행위자와 Use case를 일반화(또는 공유)할 수 있는지 평가
- 5단계 : 유스케이스 명세서 작성 (Use case description)
 - Use case 명, Actor 명 및 개요를 기술
 - 사전 및 사후 조건 (Pre- and Post condition)과 제약사항들을 식별
 - 작업(정상, 대치, 예외)흐름과 시나리오를 도출
 - 유스케이스 흐름에서 포함이나 확장 유스케이스로 구조화

Use case description

■ 개요

- Use case diagram을 보완하기 위한 산출물
- 각각의 Use case에 대해서 해당 Use case가 어떻게 수행되는지를 표현하는 수단
- Use case (UC) 상세화



Use case description

■ Use case 명세서 양식 (예)

Use case #2 : Log-in	
GENERAL CHARACTERISTICS	
Summary	자원봉사자와 사회복지기관이 VMS 사용을 위해 회원인증을 받기 위한 기능
Scope	VMS (Volunteer Management System)
Level	User level
Author	김철민
Last Update	2018. 03. 27.
Status	Analysis (Finalize)
Primary Actor	자원봉사자, 사회복지기관
Preconditions	봉사활동을 예약하려는 자원봉사자들과 사회복지기관이 VMS에 회원가입 상태여야 한다.
Trigger	VMS에 로그인하기 위해 아이디와 비밀번호를 입력한 후 회원 인증을 받으려고 할 때
Success Post Condition	자원봉사자와 사회복지기관은 VMS사용 허가를 받는다. 이후 VMS의 모든 기능을 이용할 수 있다.
Failed Post Condition	자원봉사자와 사회복지기관은 VMS를 사용하기 위한 허가를 얻지 못한다. VMS에서 봉사활동검색기능만 이용할 수 있다.

Use case description

■ Use case 명세서 양식 (예)

MAIN SUCCESS SCENARIO	
Step	Action
S	회원이 VMS에 로그인 한다.
1	이 Use case는 회원이 VMS에 로그인할 때 시작된다.
2	회원은 VMS에서 ID와 Password를 입력하고 로그인 버튼을 클릭한다.
3	VMS에 등록된 회원인지 체크해보고 등록된 회원이라면 로그인에 성공한다.
4	이 Use case는 로그인이 성공하면 끝난다.

EXTENSION SCENARIOS	
Step	Branching Action
3	3a. 아이디나 비밀번호가 잘못되어 로그인에 실패한다. ...3a1. 아이디나 비밀번호가 잘못되었다는 메시지를 보여준다. ...3a2. 아이디와 비밀번호를 입력하는 단계로 돌아간다. (Use case #2-2)

RELATED INFORMATION	
Performance	≤ 2 seconds
Frequency	회원당 하루에 평균 2번
<Concurrency>	제한 없음
Due Date	2017.04.13.

Use case description

■ Use case 명세서 항목 설명 (요약)

- Use case 이름: Use case에 대한 가장 간결한 명세로서 Use case를 통하여 제공되는 시스템의 기능을 명확한 동사구 형태로 표현해야 한다.
- Summary: 모든 이해관계자는 가장 먼저 Use case 명세서의 Summary 항목을 통해서 Use case를 이해한다.
- Actor: 해당 Use case가 동작할 때 필요한 주변 Actor를 이해할 수가 있다. Primary Actor와 (필요시) Secondary Actor를 구분하여 기록한다.

Use case description

■ Use case 명세서 항목 설명 (요약)

- Pre-condition: Use case의 시작시 만족되어야 할 조건으로서 만족되지 않으면 Use case는 시작되지 않는다.
- Post-condition: Use case의 종료시 만족해야 하는 조건으로 Use case의 정상 동작 여부에 대한 최소한의 판단 기준으로 사용될 수 있다.
- Scenario: Use case의 관련 Actor와 System 간의 상호작용에 대한 구체적인 정의를 담고 있다.
 - 기본 시나리오(Main success scenario)
 - 확장 시나리오(Extension scenario)
- Related information: 성능, 신뢰도, 보안 등 Use case와 관련된 비기능적 요구사항(Non-Functional Requirements)을 기술한다.

Use case description 항목별 작성 개요

■ Summary

- 유스케이스가 뜻하는 시스템의 기능을 간결하고 명확하게 기술한다

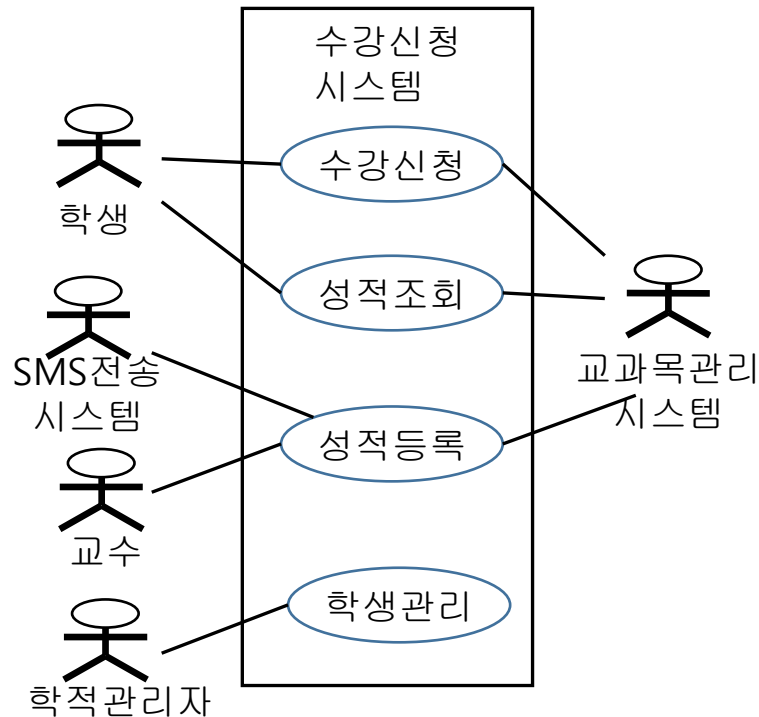
수강신청	학생은 다음 학기에 개설된 과목들 중에서 수강할 과목을 신청한다. 수강신청 기간 동안에는 신청된 과목에 대한 조회/신청취소/인쇄가 가능하다.
성적등록	교수는 담당한 각 과목 별로 수강한 학생들의 성적을 입력한다. 성적등록 기간 동안에는 입력된 성적에 대한 조회/수정/인쇄가 가능하다.
학생관리	학적관리자는 학생에 대한 정보를 등록/조회/수정/삭제할 수 있다.

엘리베이터요청	대기자는 층에서 층버튼을 눌러 엘리베이터를 요청한다. 시스템은 적절한 엘리베이터를 선택하여 요청된 층으로 엘리베이터를 이동시킨다.
목적지층으로이동	탑승자는 엘리베이터 안에서 엘리베이터버튼을 눌러 목적지 층을 선택한다. 시스템은 선택된 목적지 층으로 엘리베이터를 이동시킨다.

Use case description 항목별 작성 개요

■ Actor

- Use case와 상호작용하는 Actor 기술
 - Primary actor
 - Secondary actor



Use case 이름	관련 Actor	
수강신청	주 액터	학생
	보조 액터	교과목관리시스템
성적조회	주 액터	학생
	보조 액터	교과목관리시스템
성적등록	주 액터	교수
	보조 액터	SMS전송시스템, 교과목관리시스템
학생관리	주액터	학적관리자
	보조액터	없음

Use case description 항목별 작성 개요

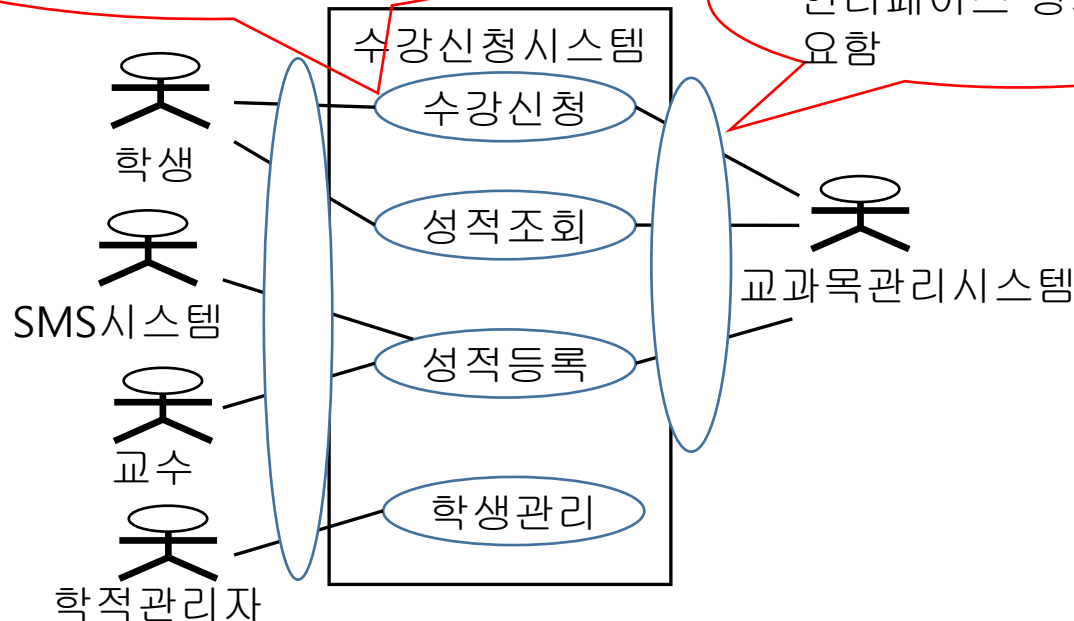
■ Actor

- Use case를 개발하기 위하여 필요한 외부 인터페이스를 파악하는데

유용

학생/교수/학적관리자와의 사용자 인터페이스, SMS시스템과의 인터페이스 정의가 중요함

교과목관리시스템과의 인터페이스 정의가 중요함



Use case description 항목별 작성 개요

■ Pre-condition (선행조건)

- Use case의 수행이 시작되기 위하여 필요한 조건을 뜻한다.
- 선행조건이 만족하지 않으면 Use case의 동작이 시작되지 않음을

Use case 이름	Pre-condition
수강신청	학생은 로그인된 상태이고 수강신청 기간이어야 한다.
성적등록	교수는 로그인된 상태이고 성적등록 기간이어야 한다.
성적조회	학생은 로그인된 상태이고 성적조회 기간이어야 한다.
도서대출신청	학생은 시스템에 로그인을 한 상태이어야 한다. 학생은 대출한도 이상을 대출한 상태가 아니다. 학생은 대출 연체료를 미납한 상태가 아니다.
엘리베이터요청	정상적으로 동작 가능한 엘리베이터가 최소한 한 개 이상 있어야 한다.
목적지층으로이동	탑승자의 엘리베이터에 장애가 없어야 한다. 즉 도착 센서, 문, 모터, 타이머가 정상 동작해야 한다.

Use case description 항목별 작성 개요

■ Post-condition (후행조건)

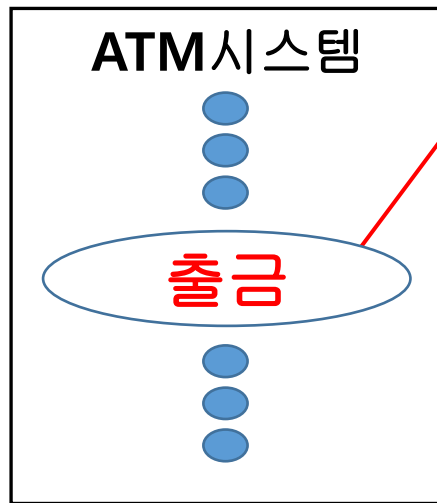
- Use case의 수행이 완료된 후에 만족되어야 하는 조건을 뜻한다.
- 해당 Use case의 정상 동작에 대한 최소한의 판단 기준이 될 수 있다.
- 만약 이 조건이 만족되지 않으면 시스템이 정상적으로 동작했다고 판단하기 어렵다.
- 그러나 후행 조건이 충족되었다고 Use case가 올바르게 수행되었다고 판단할 수는 없다.

Use case 이름	Post-condition
엘리베이터요청	Use case 종료 후의 엘리베이터 위치는 대기자가 요청한 층과 동일하다.
목적지층으로이동	Use case 종료 후의 엘리베이터 위치는 탑승자가 지정한 목적지 층과 동일하다.

Use case description 항목별 작성 개요

■ Scenario (시나리오)

- 기본 시나리오와 확장 시나리오로 구성된다.
- 기본 시나리오는 Use case에 내포된 다양한 상황 중에서 가장 일반적이고 정상적인 하나의 상황을 뜻한다.
- 확장 시나리오는 기본 시나리오가 아닌 다른 모든 시나리오를 뜻하므로 일반적이지 않은 즉 특수한 상황 혹은 비정상적인 상황을 뜻한다.

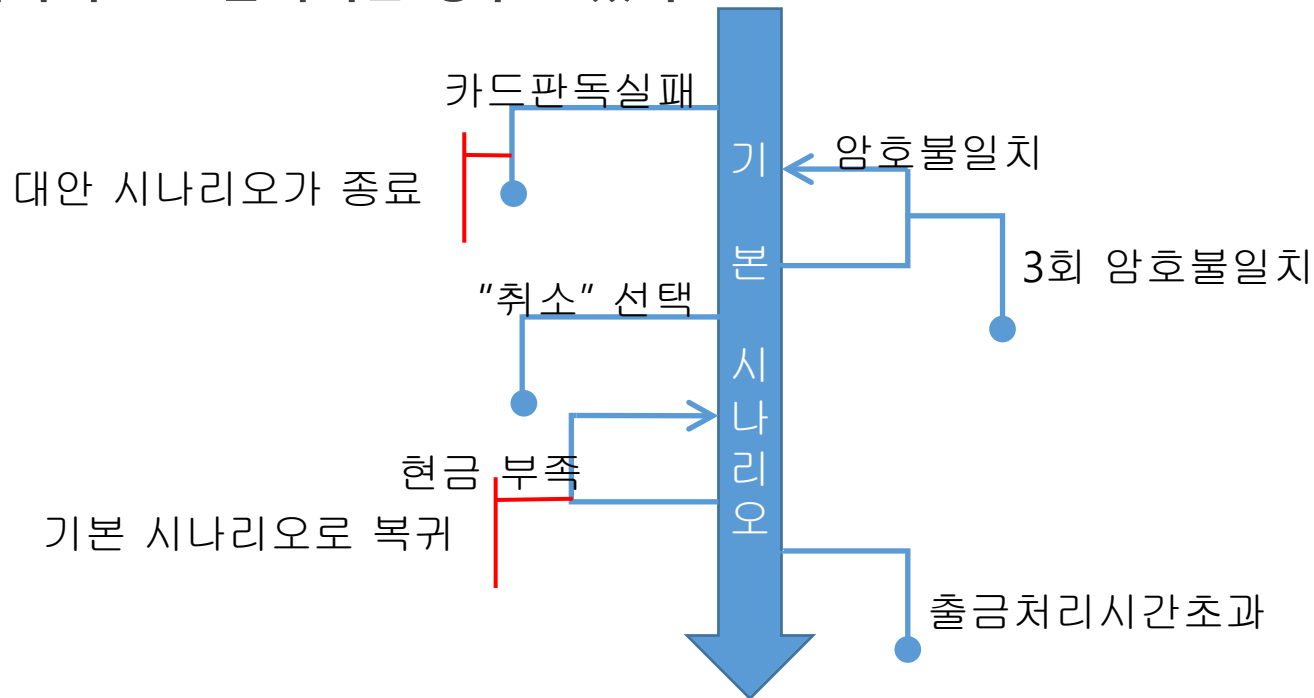


- ✓정상 출금 시나리오
- ✓카드 판독 불가 시나리오
- ✓암호 불일치 시나리오
- ✓현금 부족 시나리오
- ✓인출 취소 시나리오
- ✓출금처리시간 초과 시나리오
- ✓...

Use case description 항목별 작성 개요

■ Scenario (시나리오)

- 확장 시나리오는 기본 시나리오로부터 분기된다.
- 확장 시나리오는 분기된 후 종료되는 경우도 있고, 다시 기본 시나리오로 분기되는 경우도 있다.



Use case description 항목별 작성 개요

■ Scenario (시나리오) 작성시 유의사항

- *시나리오는 Actor와 System 간의 구체적인 상호작용을 명시적으로 정의해야 한다.
- (Ex) 출금 Use case의 기본 시나리오 명세

1. **ATM사용자**는 카드입력 장치에 카드를 삽입한다.
2. **시스템**은 삽입된 카드를 판독한다.
3. 시스템은 메뉴 화면을 출력한다.
4. ATM사용자는 "출금"을 선택한다.
5. 시스템은 암호 입력 화면을 출력한다.
6. ATM사용자는 암호를 입력한다.
7. 시스템은 입력된 암호의 정확성을 점검한다.
8. 시스템은 출금 금액 입력 화면을 출력한다.
9. ATM사용자는 인출금액을 입력한다.
10. 시스템은 **은행서버시스템**에게 출금요청을 한다.
11. 은행서버시스템은 요청된 출금에 대한 처리 결과를 시스템에게 통보한다.
12. 시스템은 카드와 지폐를 배출하고, 영수증은 인쇄한다.
13. ATM사용자는 카드, 지폐, 영수증을 수령한다.
14. 시스템은 지폐 배출 문을 닫는다.

Use case description 항목별 작성 개요

■ Scenario (시나리오) 작성시 유의사항

- (Ex) 출금 Use case의 확장 시나리오 명세

확장 시나리오
A1: 카드판독 실패 시 (2)에서 카드 판독이 안 될 때 1. 시스템은 카드 판독 실패 화면을 출력한다. 2. 시스템은 카드를 배출시킨다.
A2: 3회 암호 불일치 시 (8)에서 3회 부정확한 암호가 입력되었을 때 1. 시스템은 3회 암호가 부정확함을 출력한다. 2. 시스템은 카드를 배출시킨다.
A3: 취소 입력 시 (10)에서 "취소" 버튼을 누를 때 1. 시스템은 카드를 배출시킨다.
A4: 현금 부족 시 (11)에서 인출금액보다 적은 지폐가 있을 때 1. 최대 인출가능 금액을 출력한다. 2. 기본 시나리오 (10)으로 간다.
A5: 출금 처리 시간 초과 시 (12)에서 제한된 시간 내에 처리결과가 오지 않을 때 1. 시스템은 출금이 안됨을 출력한다. 2. 시스템을 카드를 배출시킨다

Use case description 항목별 작성 개요

■ Related information

- Use case와 관련된 비기능적 요구사항(NFR: Non-Functional Requirements)을 기술한다

Use case 이름	비기능적 요구사항의 예
출금	<ul style="list-style-type: none">●성능: 은행서버시스템에 출금요청을 한 후 최장 5초 이내에 결과를 받아야 한다.●신뢰성: 고객의 계좌, 인출 금액, ATM기기의 지폐는 무결성이 있어야 한다.●가용성: ATM기기는 1년 365일, 하루 24시간 동작해야 한다.
엘리베이터요청	<ul style="list-style-type: none">●성능: 대기자가 층 버튼을 누른 후 10초 이내에 엘리베이터가 도착해야 한다.●가용성: 엘리베이터는 1년 365일, 하루 24시간 동작해야 한다.
소장도서검색	<ul style="list-style-type: none">●성능: 도서관 이용자가 검색을 요청한 후 5초 이내에 검색 결과가 화면에 표시되어야 한다.

Summary: 검토 기준

Summary	Use case가 나타내는 전체적인 기능이 명확히 기술되어야 한다.
	Use case와 상호 작용을 하는 Actor를 기술한다.
	Use case의 일부 기능만을 뜻해서는 안 된다
	주요 시나리오(기본 및 확장)가 언급되어야 한다.
	관련 Actor가 언급되어야 한다.
	Summary에서 시스템 내부의 기능과 Actor와의 상호작용을 상세하게 기술할 필요는 없음 (아래 Scenario에서 작성하게 됨)
Actor	관련 Actor는 Use case diagram과 일관되어야 한다.
Pre-condition	Use case가 정상적으로 수행되기 위하여 가정하고 있는 상황을 표현해야 한다.
	선행 조건은 Use case의 수행 시작을 위하여 항상 만족이 되어야 하는 조건이다.
	선행 조건은 Actor와 시스템 상태에 대한 제약으로 표현된다.
	선행 조건은 Use case의 선행 조건은 사용자 인터페이스에 반영된다.
Post-condition	Use case의 수행 결과를 후행조건을 통하여 파악할 수 있어야 한다.
	후행 조건은 Use case의 수행 완료 후에 만족이 되어야 하는 조건이다.
	후행 조건은 입력과 시스템 상태의 변화에 대한 조건으로 기술된다.

Summary: 검토 기준

Scenario	<u>Use case와 관련된 모든 Actor와의 모든 상호작용을 기술해야 한다.</u>
	시나리오에 명확하고 이해가 용이한 문장 스타일로 기술해야 한다.
	개발자의 기술적인 용어를 사용하지 않고 도메인의 용어를 사용한다.
	<u>각 스텝은 주어는 System 또는 Actor를 이용하여 능동태의 문장으로 기술한다.</u>
	한 스텝에는 System 또는 하나의 Actor에 의한 기능/행위를 기술한다.
	System과 Actor와의 입/출력이 명확하게 기술되어야 한다
	각 스텝은 Actor와 System간의 입/출력 및 시스템의 기능의 궁극적인 목적을 기술한다.
	Actor가 인식할 수 없는 시스템 내부의 동작과 Actor와의 입/출력 방법을 상세하게 기술하지 않는다.
	<u>기본 시나리오와 주요 확장 시나리오 모두 기술해야 한다.</u>
NFRs	검증이 가능하도록 명확하고 구체적으로 기술해야 한다.

실습 예제 (I)

■ 게시판 개발 예제

- “글 등록하기” Use case에 대한 Use case description을 작성해보자.

- Use case 이름:
- Actor명:
- Summary:
- Scenario
 - Main success scenario
 - 1.
 - 2.
 - 3.
 - 4.
 - 5.
 - 6.
 - Extension scenario
 - 4a.
 - 4b.

실습 예제 (I)

■ 게시판 개발 예제

- Use case 이름: 글을 등록한다
- Actor명: 사용자
- Summary: 사용자는 원하는 글을 게시판에 등록한다.
- Scenario
 - Main success scenario
 - 1.로그인 Use case를 수행 한다.
 - 2.사용자는 글쓰기 버튼을 선택한다.
 - 3.시스템은 글쓰기 박스를 보여준다.
 - 4.사용자는 글쓰기 박스에 원하는 글을 작성한다.
 - 5.사용자는 등록 버튼을 선택한다.
 - 6.시스템은 글을 데이터베이스에 저장한다.
 - Extension scenario
 - 4a. 등록 취소를 수행하는 경우, 게시판 목록 조회 화면을 표시한다.
 - 4b. 정상 흐름 4에서 글쓰기 박스에 글을 쓰지 않고 등록 원하는 경우, "내용을 넣으세요" 라는 메시지를 표시한다.

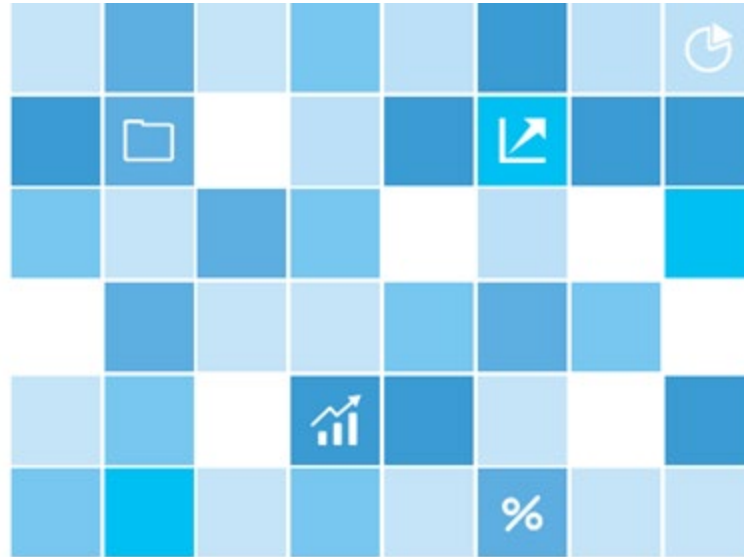
실습 예제 (III)

■ 은행 시스템 개발시

- “Withdraw cash” Use case 에 대 한 Use case description 을 작성해보자

실습 예제 (III)

유스케이스명	인출
액터명	주 액터: 사용자 부 액터: 인증시스템, 은행시스템
개요	사용자가 자신의 계좌로부터 예금 출금을 하기 위해 ATM을 이용한다.
사전조건	<ul style="list-style-type: none"> • 사용자가 은행 카드를 소지하고 있어야 한다. • 은행시스템과 인증시스템과의 네트워크 연결이 정상적이어야 한다. • 시스템은 사용자가 요구한 출금액만큼의 현금을 가지고 있어야 한다.
사후조건	<ul style="list-style-type: none"> • 사용자가 카드를 돌려받는다. 사용자가 돈을 출금하고 은행계좌에 출금액이 반영된다. • 사용자가 카드를 돌려받는다. 사용자가 돈을 출금하지 못하고 은행계좌는 변동이 없다. • 사용자가 카드를 돌려받지 못한다. 사용자가 돈을 출금하지 못하고 은행계좌는 변동이 없다. 또한 은행을 접촉하라는 경고를 받는다.
기본흐름	<ol style="list-style-type: none"> 1. 사용자는 은행 카드를 시스템에 제시한다.(트리거) 2. 시스템은 카드 정보를 읽고 비밀번호를 요구한다. 3. 사용자는 비밀번호를 입력한다. 4. 「사용자인증」 유스케이스를 실행한다. 5. 시스템은 사용자에게 트랜잭션 타입을 요구한다. 6. 사용자는 출금 트랜잭션을 선택한다. 7. 시스템은 출금액을 사용자에게 요구한다. 8. 사용자는 출금액을 입력한다. 9. 시스템은 은행시스템에 출금을 요구한다. 은행시스템은 사용자의 계좌에 출금을 반영한다. 10. 시스템은 돈을 사용자에게 내주고 출금 사실을 로그기록에 반영 한다 11. 시스템은 은행카드를 사용자에게 되돌려 준다.
대체흐름 1	2a. 시스템이 인식하지 못하는 카드가 입력되는 경우 2a.1 시스템은 사용자에게 카드 판별할 수 없다는 사실을 알리고 유스케이스를 종료한다.
대체흐름 2	9a. 잔금이 모자라는 경우 9a.1 사용자에게 알리고 단계 7을 다시 수행한다.
대체흐름 3	...



Thank You