

사용 사례 Use Cases

- 사용 사례는 시스템을 사용하는 방식을 특징적으로 나타낸 것
- 사용자와 시스템 사이의 다이얼로그를 사용자의 관점에서 나타낸 것
- 기능적 요구를 캡처 한 것.

사용 사례 작성 이점

- 시스템 개발자와 고객 사이에 요구를 이해하는 수단이 됨(정상적인 시나리오)
- 예외적인 케이스를 개발자에게 주지시킴(확장 시나리오)
- 대략적인 계획을 위하여 기능의 수준을 파악(목표 리스트)



용 어

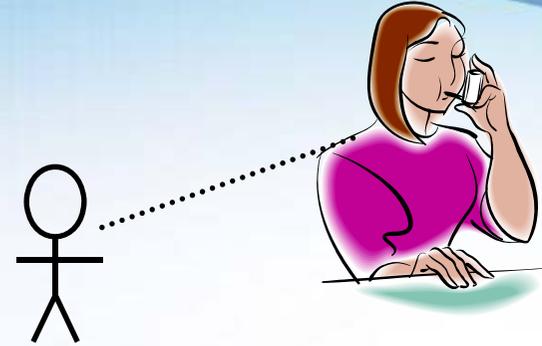
- 액터 : 시스템과 상호 대화하며 사용하는 사람
- 주 액터(primary actor) : 액션을 구동시키는 사람
- 목표(goal) : 주 액터가 달성하는 성과
- 수준 :
 - 요약 목표
 - 사용자 목표
 - 서브 기능

액터

□ 액터 인스턴스

- 액터는 사용자의 Role
- 구체적인 사용자는 여러 명

□ 사용자가 여러 액터로 역할 가능



도서관 열람자



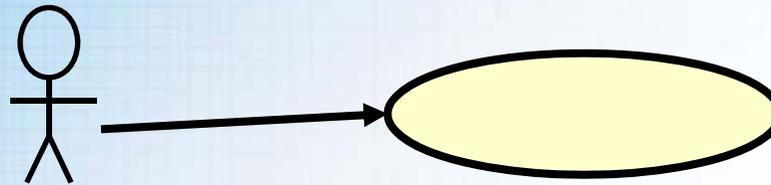
도서관 사서



사용사례

- 사용사례는 개별 시나리오의 모임

- > 정상적 흐름
- > 오류
- > 예외



도서관 사서

연체 통보

시나리오 #1

연체 통보 메뉴 선택
시스템 기간 선택창 디스플레이
기간 선택
연체자 리스트 디스플레이
발송 선택

시나리오 #2

연체 통보 메뉴 선택
시스템 기간 선택창 디스플레이
기간 선택
기간 선택 오류
연체자 리스트 디스플레이
발송 선택

사용 사례에 표현하는 것들

□ 다음 요구사항들 중 사용 사례에 직접 표현되는 것은?

1. 주문 총액 = 주문 아이템 비용 * 1.06 부가가치세금
2. 이벤트 판매는 6개월 이상 지속되지 않음.
3. 1년 이상 된 고객만이 프리미엄 회원이 됨.
4. 고객은 1회의 구매 상담이 가능함.
5. 반응 시간은 2초 이내.
6. 가동시간은 99.8%.
7. 동시 사용자 수 최대 200명.

사용 사례의 스타일

□ 사용 사례 다이어그램

○ 시스템의 모든 사용 사례를 보여주는 개요

□ 비정형 사용 사례

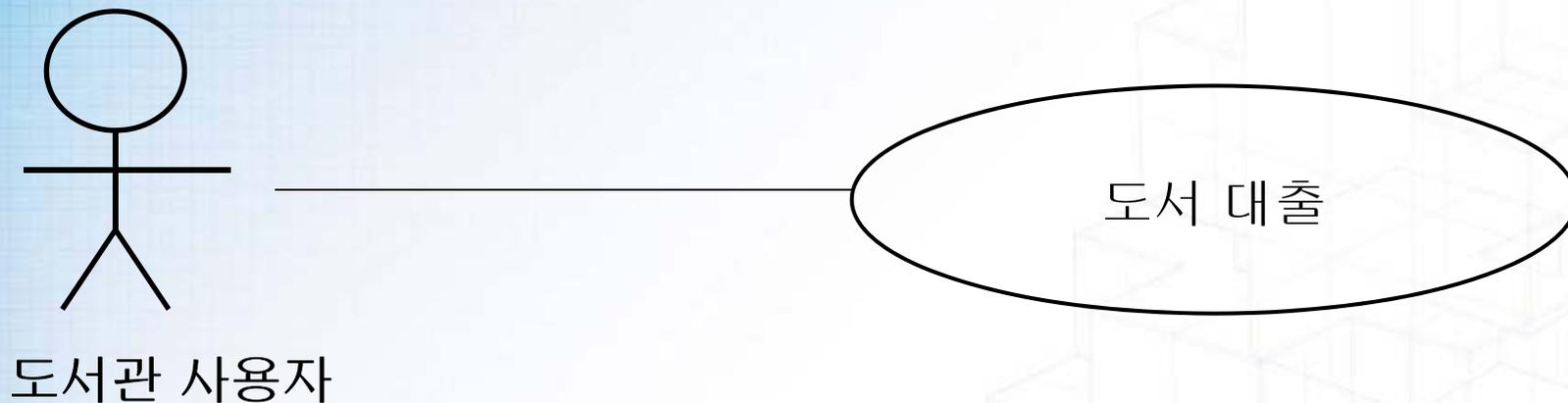
□ 정형 사용 사례

하나씩 상세히 살펴보면 ...

1. 사용 사례 다이어그램

□ 시스템에 있는 사용 사례 모두를 나타낸 다이어그램(요약 다이어그램)

- 스틱맨으로 표시한 액터(명사형의 이름)
- 타원형으로 표시한 사용 사례(동사형의 이름)
- 액터와 사용 사례를 연결한 관계 선
- 사용 사례가 사용 또는 의존하는 다른 사용 사례와 연결

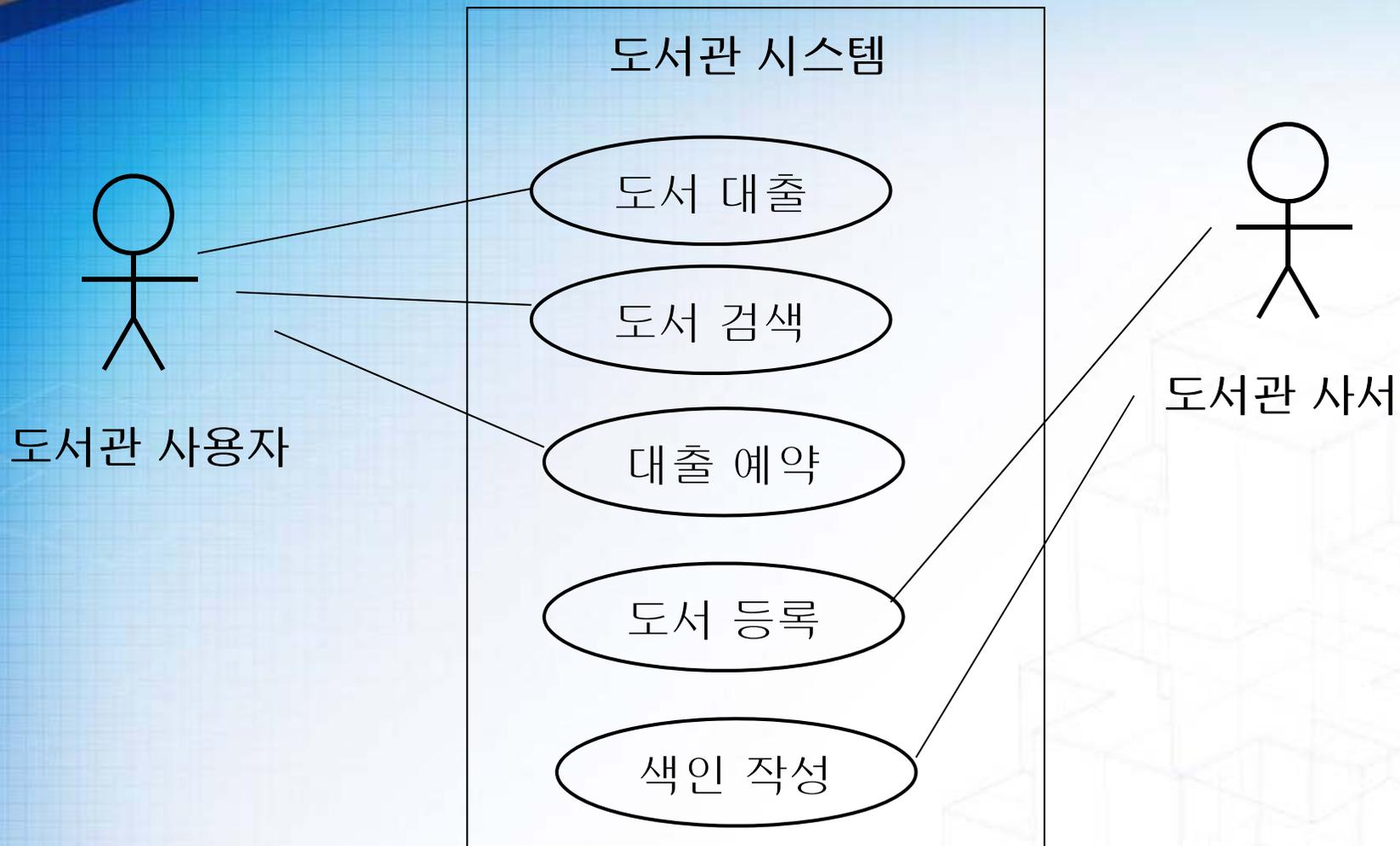


사용 사례 요약 다이어그램

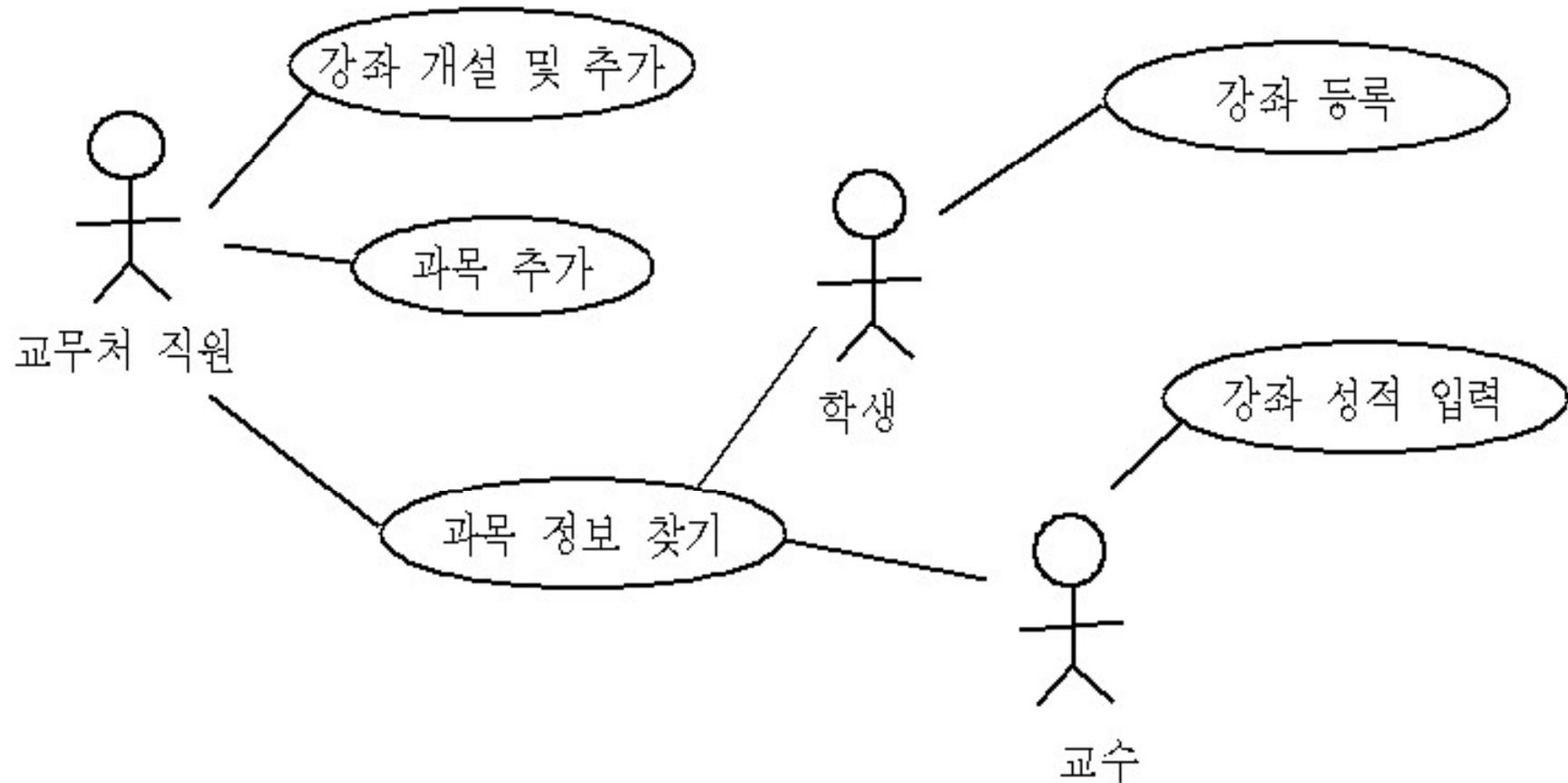
- 주 액터와 (액터가 구동시키는) 목표의 리스트를 표 형태로 만들면 유용함.

액터	목표
도서관 사용자	책 검색
	책 대출
	책 반납
도서관 사서	책 검색
	도서
	타 도서관 열람 요청

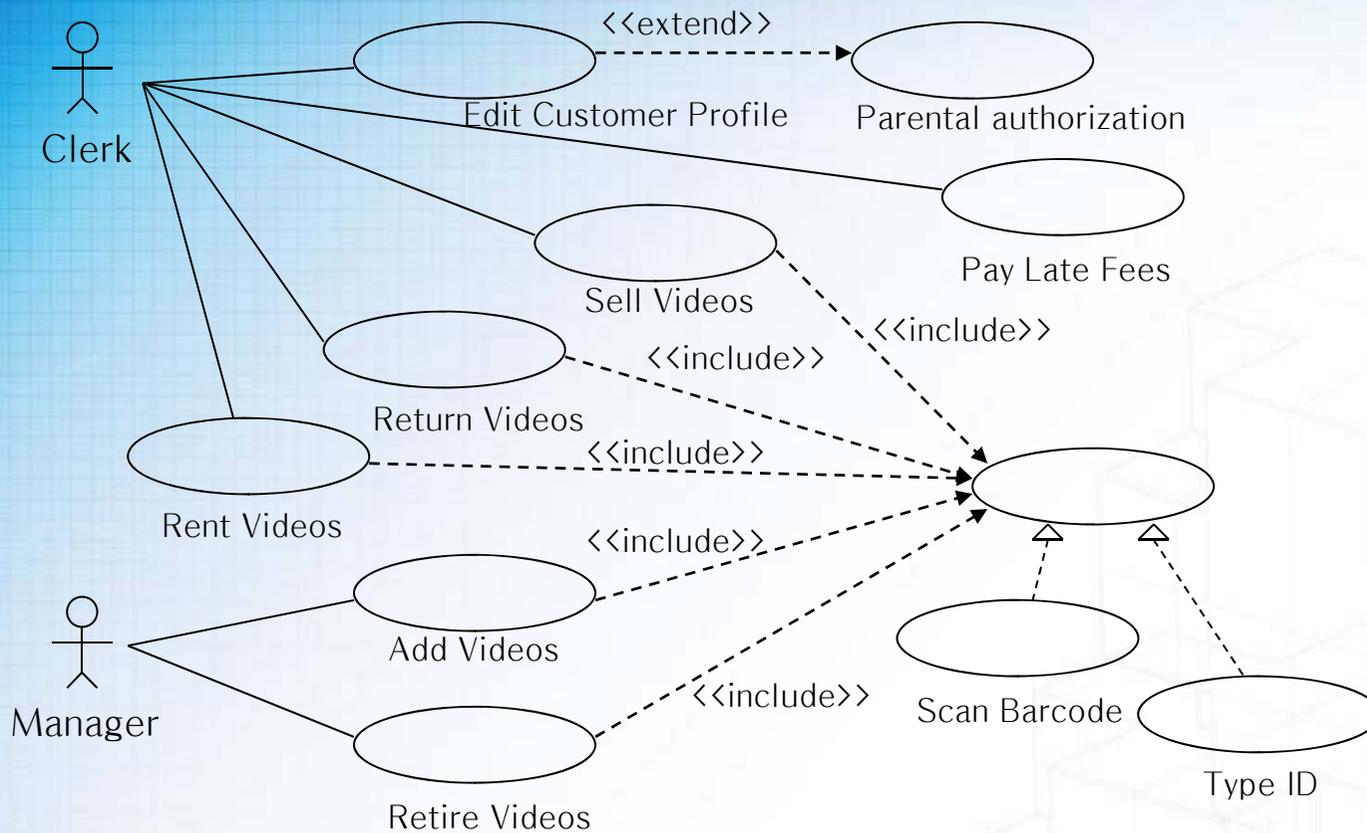
사용 사례 요약 다이어그램 1



사용 사례 요약 다이어그램 2



사용 사례 요약 다이어그램 3



확장과 포함관계

□ 확장관계 <<extends>>

- 예외 사항을 나타내는 관계
- 이벤트를 추가하여 다른 사례로 확장
- 시작 조건과 종료 조건 필요

□ 포함관계 <<includes>>

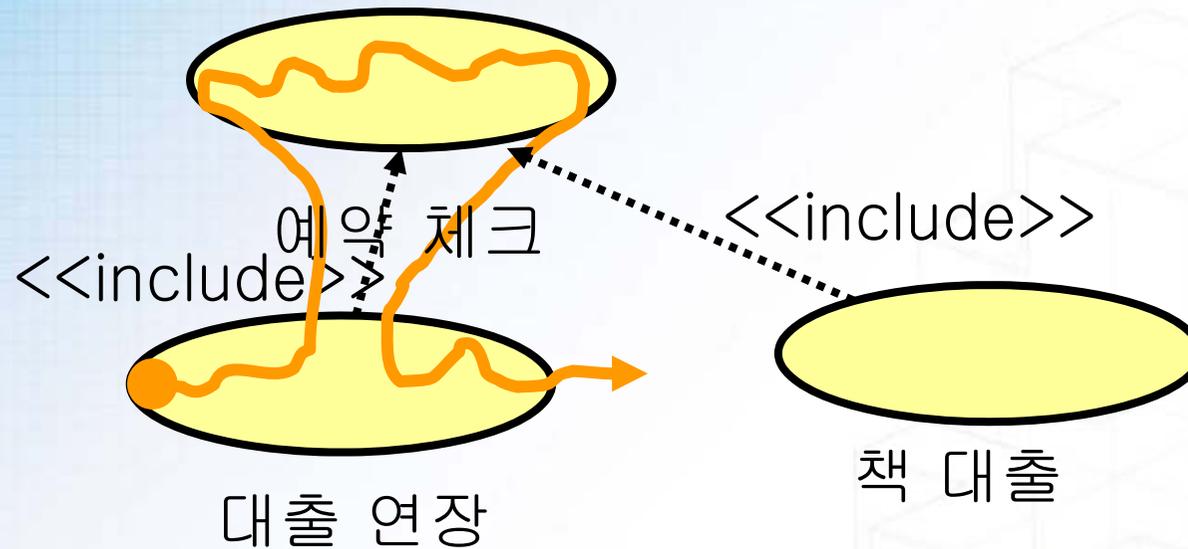
- 사용사례 사이의 중복 제거
- 함수의 호출처럼 포함된 사용사례를 호출하는 의미

□ 확장관계와 포함관계의 차이

- 관계의 방향(화살표 방향)
- 포함관계 - 목표 사용사례가 구동되는 조건이 출발 사용사례에 기술
- 확장관계 - 확장이 적용되는 조건이 확장 사용사례의 시작조건으로 기술

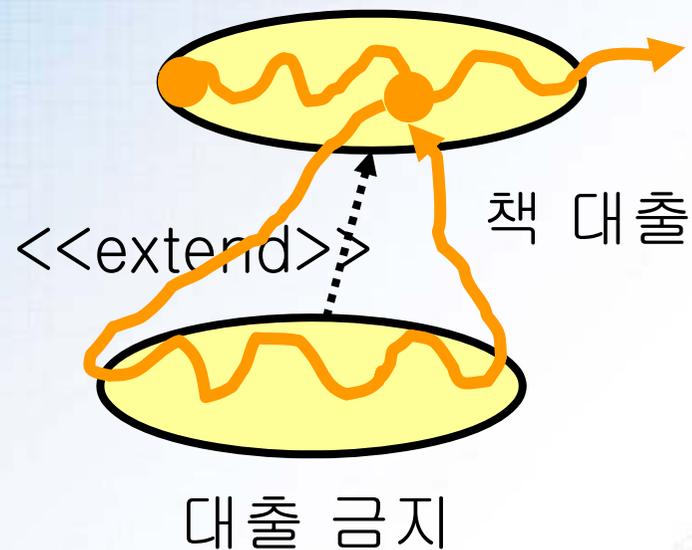
포함 관계

- 사용사례 사이의 중복 제거
- 함수의 호출처럼 포함된 사용사례를 호출하는 의미



확장 관계

- 예외 사항을 나타내는 관계
- 이벤트를 추가하여 다른 사례로 확장
- 예외 시작 조건과 종료 조건 필요



2. 비정형 사용 사례

□ 비정형 사용 사례는 시나리오/인터랙션을 글로 쓴 문단.

○ 사례: 책 분실

도서관 사용자가 책을 분실하였다는 사서에게 보고한다. 사서가 도서관의 기록을 인쇄하고 사용자에게 배상금을 요청한다. 배상금은 책임 사서가 결정한다. 시스템은 책이 분실되었음을 반영하고 사용자의 대출 기록을 변경한다. 책임 사서는 대체할 새 책을 주문한다.

3. 정형 사용 사례

목표	도서관 사용자가 온라인 색인을 이용하여 책 대출을 예약하기를 원함.
주 액터	도서관 이용자
범위	도서관 시스템
수준	사용자
사전조건	로그인 상태
성공조건	책이 예약됨
실패조건	책이 예약되지 않음
구동	사용자가 시스템에 로그인

<p>주요 성공 시나리오</p>	<ol style="list-style-type: none"> 1. 사용자가 계정 아이디와 패스워드 입력 2. 시스템이 아이디를 검증하고 로그인 3. 시스템이 탐색 화면과 함께 색인 제공 4. 사용자가 책 이름 입력 5. 시스템이 일치하는 책을 찾아 선택할 수 있는 리스트 제공 6. 사용자가 책을 선택하고 예약 7. 시스템이 예약을 확인하고 색인에 반영
<p>확장(오류 시나리오)</p>	<ol style="list-style-type: none"> 2a. 패스워드가 정확하지 않음. <ul style="list-style-type: none"> 2a.1 시스템이 로그인 화면으로 리턴 2a.2 사용자가 다시 로그인 시도 5a. 시스템이 책을 발견할 수 없음. <ul style="list-style-type: none"> 5a.1 ...
<p>변형 (선택 시나리오)</p>	<ol style="list-style-type: none"> 4'. 사용자가 저자 또는 주제를 입력

사용 사례를 작성하는 순서

1. 액터와 목표를 식별

시스템을 구동시키는 컴퓨터, 서브 시스템, 사람이 무엇인가?(액터)

각 액터가 시스템 무엇을 하기를 원하는가?(목표)

2. 성공 시나리오 작성

이벤트의 흐름을 작성

3. 대안 시나리오 작성

확장이나 변형 시나리오 작성

2. 성공 시나리오의 작성

- 성공 시나리오는 “해피 경로”
 - 가장 읽기 쉽고 이해하기 쉬운
 - 복잡한 것은 따로 기술된

- 처음부터 완료될 때까지 액터의 의도와 책임을 담아 놓은 것
 - 어떤 정보들이 액터와 시스템 사이에 오가는지
 - 한 줄 한 줄 작성

3. 오류 확장 시나리오의 작성

- 대체로 모든 이벤트는 실패가 발생할 수 있음.
- 중요한 것은 실패 조건을 메인 성공 시나리오 다음에 기술
- 이벤트 번호와 문자를 붙임
 - 5.a 실패 조건
 - 5.a.1 실패가 일어났을 때의 사용 사례
 - 5.a.2 계속...
- 연습: 도서관 사용자가 책을 탐색하려 하였으나 찾을 수 없는 경우

4. 변형

- 많은 이벤트가 변형적인 동작과 시나리오를 가짐.
- 스텝 번호와 변형으로 표시
 - 5'. 스텝 5의 변형 1
 - 5''. 스텝 5의 변형 2

- 연습: 도서관 사용자가 저자를 입력한 경우 일어난 변형은?

좋은 사용 사례의 조건

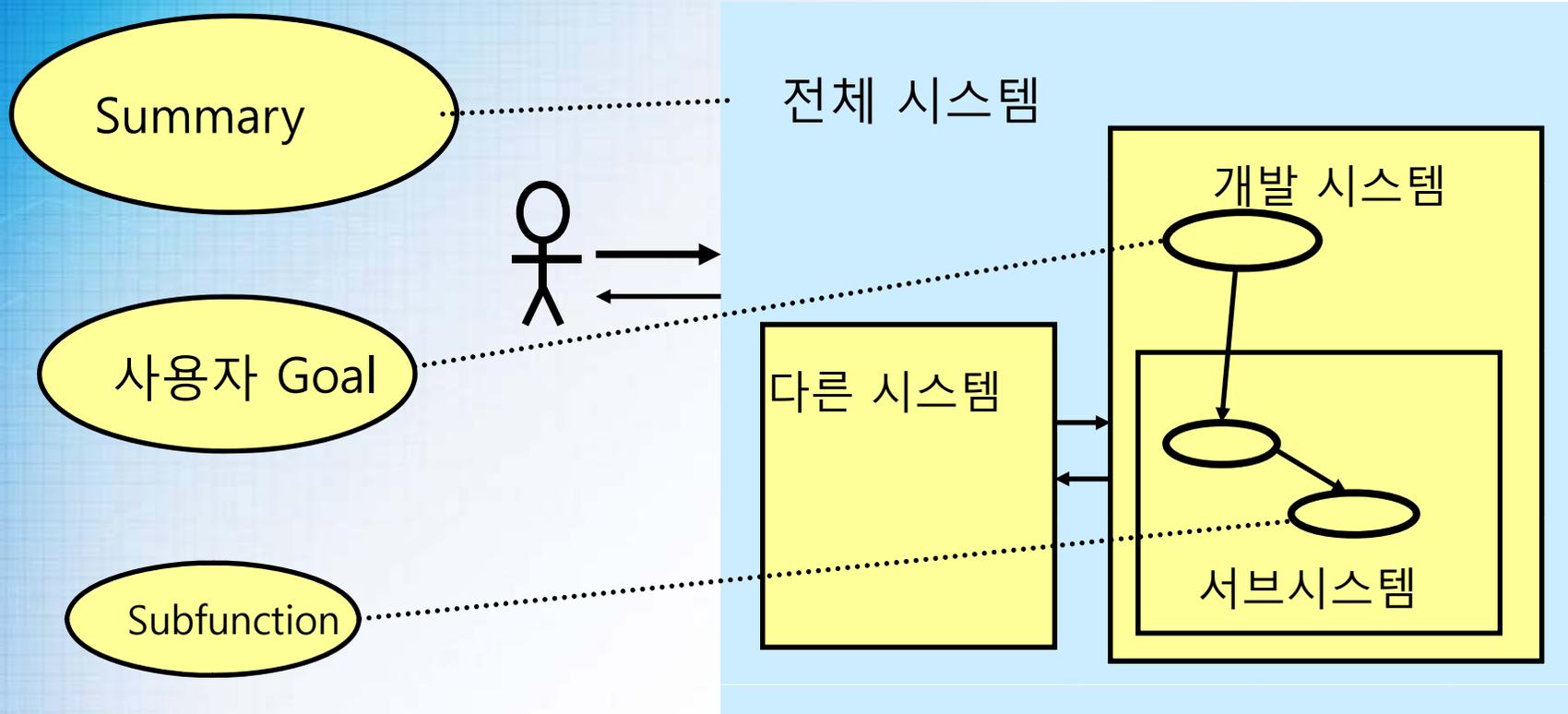
□ 좋은 사용 사례는

- 액터로부터 시스템에 서비스의 요청으로 시작하여
- 요청한 모든 답을 얻은 후 종료
- 시스템과 액터 사이의 인터랙션을 정의
- 시스템의 관점이 아니라 액터의 관점에서 기술
- 시스템 내부의 동작이 아니라 상호작용에 초점
- GUI를 자세히 기술할 필요는 없음
- 메인 성공 시나리오는 3-9 단계
- 읽기 쉬울 것

□ 좋은 사용 사례 요약은 한 페이지에

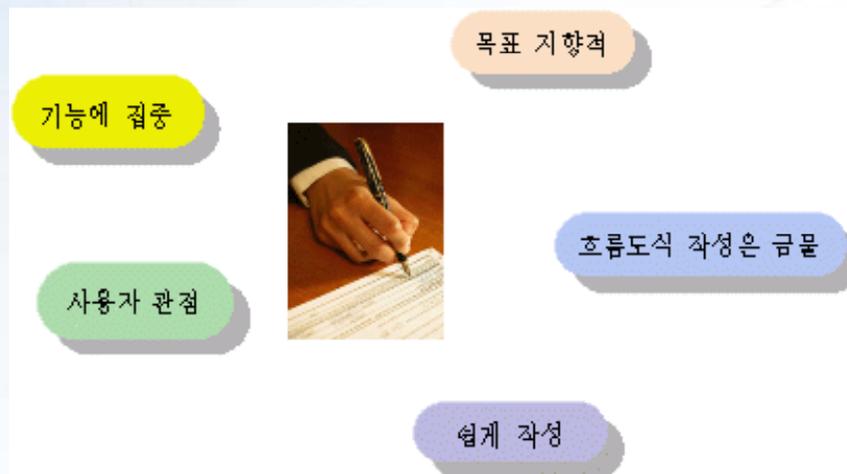
사용 사례의 규모와 수준

- 사용 사례는 얼마만한 크기?
사용자 Goal - 외부에 보여지는 기능 단위



사용사례 다이어그램 작성시 유의점

- 소프트웨어로 제공될 기능에 집중
- 사용자의 관점으로 작성하라
- Goal 지향적
- 시스템이 제공하는 서비스에 초점
- Data Flow Diagram과는 다름
- 논리의 흐름(순차, 선택, 반복)이 직접 표시될 수 없음



사용 사례를 이용하면....

- 시스템의 범위를 정하는데 도움이 됨
- 개발 과정을 계획하는데도 사용됨
- 요구를 개발하고 검증하는데 사용 됨
- 테스트케이스를 정의하는데 기초가 됨
- 사용자 매뉴얼 구성하는데 사용될 수 있음

실습: 사용 사례 프로젝트

- 프로젝트 팀을 두 그룹으로 나누어
- 각 그룹이 프로젝트를 위한 사용 사례 작성
- 수업 시간에 발표하기 위하여 하나의 팀 결과를 선택



요구 명세서에 지금 작업한 사용 사례를 포함

사용 사례 취합

어느 정도 자세해야 하나?

□ 균형을 잘 유지해야

- 이해 가능 vs. 상세함
- 그래픽 vs. 상세한 글과 표
- 짧고 빠르게 vs. 완벽하고 느리게

고객, 관계, 융통성에 따라 균형이 달라질 수 있음.

사용 사례 작성 팁

명세서 작성 후 다음 사항을 다시 훑어 본다.

- 꼭 필요하지 않는 요구는 삭제
- 필요 이상으로 복잡한 것은 단순화
- 더 저렴한 옵션이 있다면 교체
- 필수적이 아닌 것은 미래의 릴리스로



사용 사례 작성 팁

애자일 방법의 원리 - 단순화가 관건

