

유스케이스 모델링

(Use case Modeling)

유스케이스 모델링(Use case Modeling)

- 유스케이스(Use case)
 - 사용자 입장에서 시스템의 기능을 기술한 시나리오
- 유스케이스 다이어그램(Usecase Diagram)
 - UML 다이어그램의 하나로 시스템 외부의 액터와 시스템이 제공하는 기능들을 구조적으로 표현한 다이어그램
- 요구사항 분석
 - 자연어로 작성된 요구사항이 정확하고 완전하며 일관성이 있는지 검토하여 수정하는 작업
- 유스케이스 모델링
 - 자연어로 구성된 요구사항을 유스케이스 다이어그램을 사용하여 구조화한 후 이것을 정형화하고 명세화하는 작업
 - 유스케이스 다이어그램을 작성하면서 전체 시스템을 추상화하여 파악할 수 있고 액터와 유스케이스 간의 관계를 통해 개발해야 하는 시스템의 범위를 명확히 할 수 있다.
 - 유스케이스 명세는 요구사항을 구조화한 것으로 보다 정형적인 분석을 가능하게 하고 고객과의 원활한 의사소통을 위한 수단을 제공한다.

유스케이스 모델링(Use case Modeling)

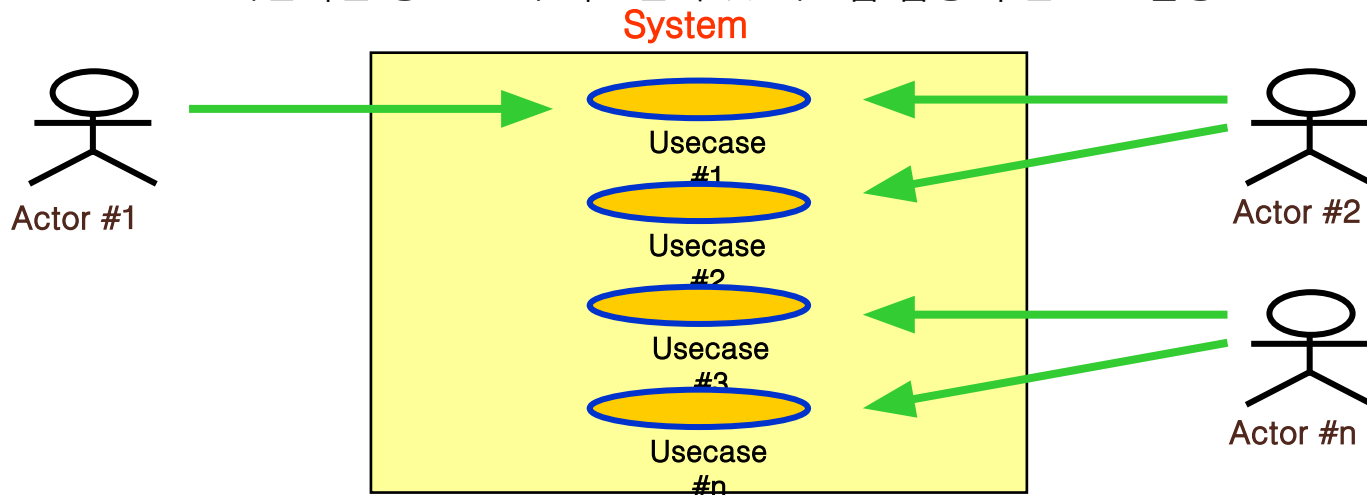
1. Usecase Diagram

1) Definition

- 고객의 요구사항을 구체화하는 작업으로 **사용자의 관점에서 시스템이 무엇을 수행해야 하는지의 기능(Use case)과 시스템과 관련된 외부요소(Actor)와의 상호작용을 도형으로 표현**






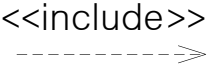
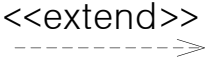
2) Characteristics

- Usecase Diagram은 최종사용자나 비즈니스전문가, 개발자 간의 의사소통 수단으로 활용
- Usecase Diagram은 시스템에 대한 쉬운 이해를 도모하여 요구사항 식별 수단은 물론 이를 명확화하여 개발을 쉽게 함
- Usecase는 개발기간 중 Test의 기본단위 및 시스템 검증 수단으로 활용



유스케이스 모델링(Use case Modeling)

2) Elements & Relationship

Classification	Construct	Description	Syntax
Elements	actor	A coherent set of roles that users of use cases play when interacting with these use cases.	
	use case	A sequence of actions, including variants, that a system (or other entity) can perform, interacting with actors of the system.	
	system boundary	Represents the boundary between the physical system and the actors who interact with the physical system.	
Relationships	Generalization	A taxonomic relationship between a more general use case and a more specific use case.	
	association	The participation of an actor in a use case. i.e., instance of an actor and instances of a use case communicate with each other.	
	include	An relationship from a base use case to an inclusion use case, specifying how the behavior for the inclusion use case is inserted into the behavior defined for the base use case.	
	extend	A relationship from an <i>extension</i> use case to a <i>base</i> use case, specifying how the behavior for the extension use case can be inserted into the behavior defined for the base use case.	

유스케이스 모델링(Use case Modeling)

3) Actor (1/2)

- Definition

- 시스템과 상호작용하는 사람이나 외부 시스템을 의미한다.
- 시스템을 시작시키거나 시스템의 기능을 활용하여 혜택을 받는 객체이다.

- Notation

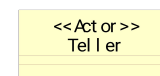
- 일반적으로 “Stick Man”으로 표시 : 주로 Usecase Diagram에서 표시
- Stereotype이 “actor”인 Class로 표시 가능 : 주로 Class Diagram에서 표시



User

- Actor 유형

- 주요 Actor : 시스템의 서비스를 이용해 목적 달성
- 지원 Actor : 시스템에 서비스를 제공



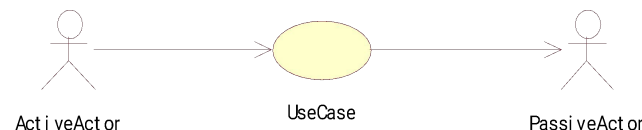
- Active Actor vs Passive Actor

- Actor와 Usecase간의 Direction에 따라 구분
- Active Actor

- » Usecase에게 메시지를 보내는 Actor
- » 사람, 외부 시스템, 외부 디바이스, 객체 등이 될 수 있음

- Passive Actor

- » Usecase로부터 메시지(결과)를 받는 Actor
- » 외부 시스템, 외부 디바이스 등이 될 수 있으나, 사람이 Passive Actor로 되는 것에는 냉철한 판단 필요



유스케이스 모델링(Use case Modeling)

3) Actor (2/2)

- Actor 추출법

- 누가 이 시스템을 시작하고 끝내는가?
- 누가 개발될 시스템의 주요기능을 이용하는가?
- 누가 시스템이 만들어낸 결과에 관심이 있는가?
- 누가 정보를 제공하고 사용하고 제거 하는가?
- 누가 어떤 요구사항에 관심 있어 하는가?
- 누가 시스템을 통해 일상 업무를 지원 받는가?
- 어떤 팀에서 시스템을 사용하는가?
- 개발할 시스템과 연동되는 다른 시스템에는 어떤 것이 있는가?
 - » 기존 시스템 포함
- 누가 외부 자원을 사용하는가?
- 특정 한 사람이 다른 여러 역할을 수행하는가?
- 특정 여러 사람이 동일한 역할을 수행하는가?
- 누가 시스템이 잘 운영될 수 있게 유지보수 및 관리를 하는가?

유스케이스 모델링(Use case Modeling)

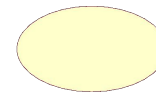
4) Usecase (1/3)

- Definition

- Actor에 의해 수행이 시작되어 Actor가 궁극적으로 원하는 결과를 내는 시스템이 수행하는 최소의 기능단위
- Usecase는 시스템이 나타내는 행위의 한 패턴
 - » 하나의 Usecase는 일련의 액션들로 표현
- Usecase는 Actor에 의해 수행이 시작되고, Actor가 관찰할 수 있는 결과를 산출
- 시스템 사용자 입장에서의 기능적인 요구사항
- 외부시스템과 상호작용하는 행위들의 집합
- 시스템 또는 서브시스템의 필수적인 행동 및 범위
- 요구사항 추출 및 최종사용자 혹은 고객과의 의사소통 수단
- Usecase는 대상 행동을 명세화만을 수행하고 행동 수행 방법은 규정하지는 않음
- 하나의 Usecase는 여러 Variants를 포함하고 있다
- Usecase Diagram 과 Usecase Description 으로 구성

- Notation

- Name을 포함한 타원으로 표시
- Name은 Path(Package)를 포함한 Name으로 표시가능
- 각 Usecase는 시스템과의 Interaction sequence



Make a Reservation

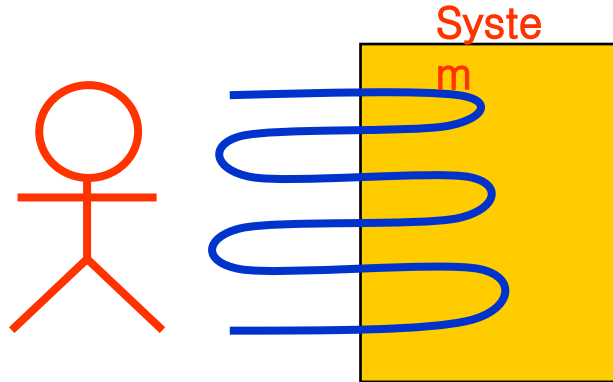


Find Room
(from Room)

유스케이스 모델링(Use case Modeling)

4) Usecase (2/3)

- Usecase vs Function



- Usecase 추출법

- 사용자 Actor가 시스템이 제공하기 원하는 기능은 무엇인가?
- 시스템이 어떤 기능을 제공하면 사용자 Actor의 일상 작업이 효율적이고 편해지는가?
- 사용자 Actor는 시스템에 어떤 정보를 생성, 수정, 조회, 삭제하고 싶어 하는가?
- 시스템의 입력과 출력은 무엇인가?
- Actor는 시스템의 갑작스런 외부 변화에 대해 어떤 정보를 필요로 하는가?
- Actor가 한 event에 대해 시스템이 어떤 정보를 알려줘야 하고 반대로 하나의 event를 위해 Actor가 시스템에 어떠한 정보를 줘야 하는가?
- 어떤 Usecase가 시스템을 지원 또는 유지보수 할 것인가?

- Usecase 와 Actor

- Actor가 Usecase의 수행을 시작 시키고, 수행결과를 통보 받음
- Actor는 Usecase 기능완료를 위해 자신의 기능을 시스템에 제공
- Actor는 Usecase와 교류할 때 Usecase 사용자들의 역할을 표현
- Actor는 Usecase에 연관 관계를 이용하여 연결 표현

유스케이스 모델링(Use case Modeling)

4) Usecase (3/3)

- Good Usecase

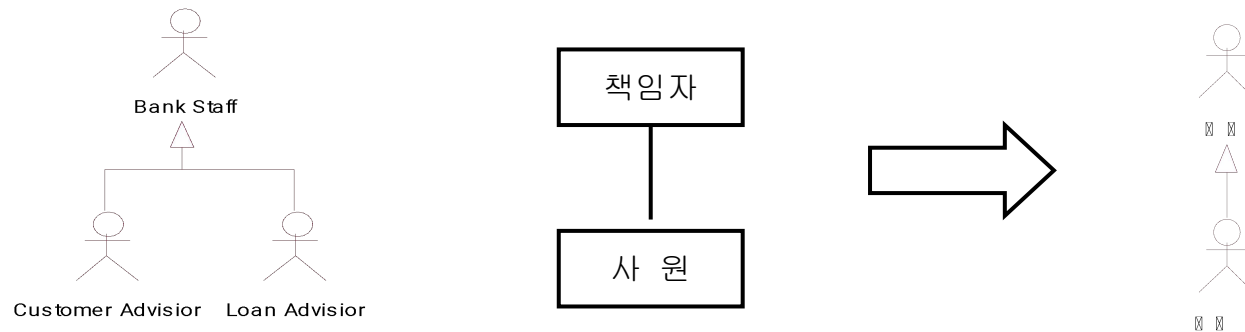
- Good Usecase는 주요 기능을 시작부터 끝까지 완벽하게 제공
- System의 정적인 Usecase 관점에서 한 부분의 대화에만 초점을 맞춤
- 해당 부분을 이해하는데 필수적인 Usecase와 Actor만을 포함
- 추상화 수준별로 일관성 있게 상세 사항을 추가
- 중요한 의미는 적절한 정도의 크기로 분해
- Usecase 명칭은 단일 명칭, 식별 가능 명칭 사용
- Usecase가 포함하고 있는 행동을 공통 Usecase로 분해 : Include
- 변이성 행동을 구별하여 다른 Usecase에 확장을 표현 : Extend
- Event 흐름을 명확히 하여 쉽게 알 수 있어야 함
- Usecase 의미가 정상인 것과 변이적인 것을 초소한의 Scenario로 명세화

유스케이스 모델링(Use case Modeling)

5) Relationship (1/3)

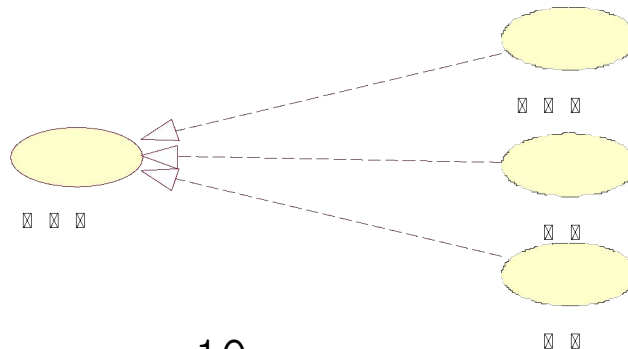
- Generalization

- Actor간 또는 Usecase간의 관계를 나타낼 때 사용하며 상속의 의미를 내포



- Realization

- Collaboration과 Usecase간의 실체화 관계를 나타낼 때 사용
 - » 실제 프로젝트에서는 자주 사용되지 않는 개념임
 - » Requirement 단계의 Usecase가 실체화 될 때 Collaboration으로 Realize를 사용

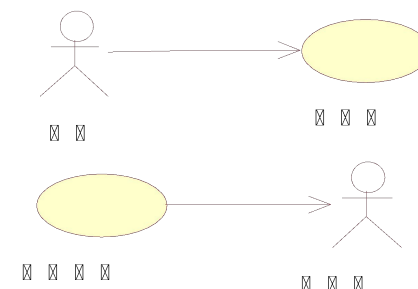


유스케이스 모델링(Use case Modeling)

5) Relationship (2/3)

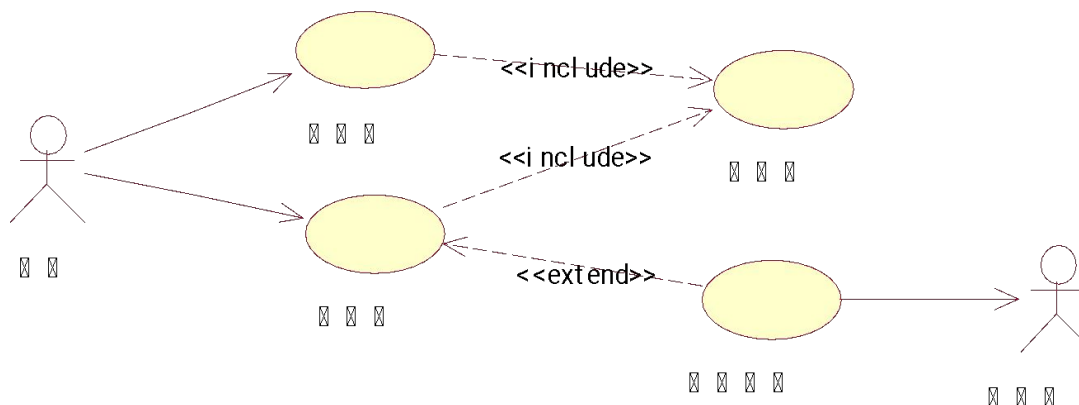
- Association

- Actor와 Usecase사이의 관계를 나타낼 때 사용
- Communicate의 의미를 내포
- Direction은 메시지가 전달되는 방향을 의미
 - » Direction에 따라 Active/Passive Actor로 구분



- Dependency (1/2)

- Usecase와 Usecase사이의 관계를 표현하며, 주로 포함(Include)과 확장(Extend)의 의미를 가짐
- Usecase간의 Dependency는 Stereotype을 이용하여 그 관계를 명확하게 밝혀 주는 것이 일반적
 - » Include/Extend Stereotype을 사용



유스케이스 모델링(Use case Modeling)

5) Relationship (3/3)

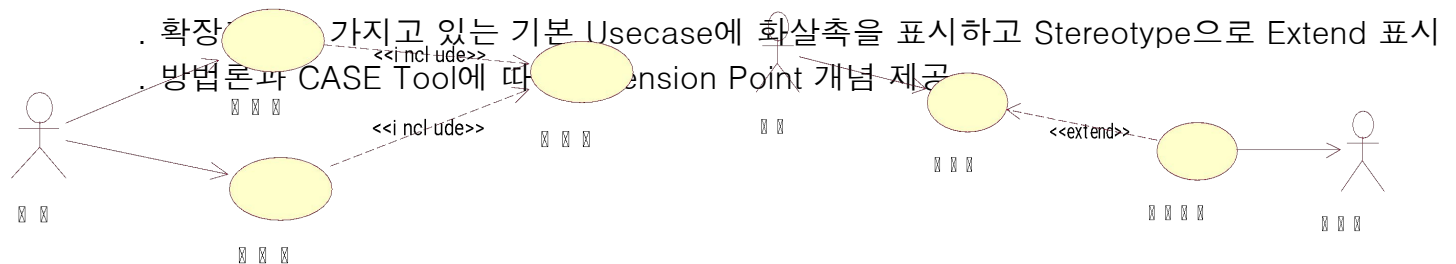
- Dependency (2/2)

- Include Stereotype

- » Usecase의 이벤트 흐름 안에 다른 Usecase의 행동을 포함
- » Usecase 간에 공통 발생 부분을 뽑아내서 참조
- » Including Usecase는 Included Usecase 없이 제구실을 못함
- » 비밀번호 확인, 고객 확인 등 재사용이 가능한 모듈과 같은 기능에 대해 사용
- » 일반적으로 Include 된 Usecase는 Actor로부터 Communicate되지 않음
- » Notation : 사용하는 Usecase 방향으로 화살촉을 표시하고 Stereotype으로 Include 표시

- Extend Stereotype

- » 기본 Usecase에서 Optional한 어떠한 조건에 의해 확장된 기능을 제공하는 Usecase가 수행되는 경우 사용
- » 한 Usecase의 인스턴스가 다른 Usecase 기능에 의해 확장
- » 확장시키는 Usecase를 별도로 하여 관리를 용이하게 함
- » 확장되는 Usecase의 수정 없이 기능 추가 가능
- » Extended Usecase는 extending Usecase없이도 제구실을 함
- » 의미적으로는 Generalization과 유사하지만, Extend Usecase가 실행되는 조건에 제약을 부여하고자 하는 경우 사용
- » Notation



유스케이스 모델링(Use case Modeling)

6) Usecase Diagram 작성 (1/2)

- 작성 절차

- 시스템 경계를 선택

- » 외부의 주요 Actor 및 지원 Actor 찾기
 - » “무엇이 외부에 있는가?”의 시스템 경계 식별

- 시스템과 교류하는 모든 Actor들을 식별한다.

- » Actor중 일반적인 혹은 더 특수화된 Actor 들을 식별하여 Actor를 조직, 구성
 - » Actor별 사용자 목적 식별
 - » 각 Actor에 관한 Interaction에서 중요한 것을 파악하여 Usecase를 추출한다.

- 사용자 목적을 만족하는 Usecase 정의

- » Interaction을 통한 상태의 변화를 Usecase로 추출한다.
 - » 추출된 Usecase에서 좀 더 일반적인 혹은 특수화된 Usecase로 구성하여 공통 행동은 분해하고 예외 행동을 파악한다. : Include/Extend

- Usecase Diagram 작성지침 (1/2)

- 목적을 알 수 있는 명칭의 사용
 - 관계는 너무 많이 표현하지 않고 반드시 필요한 것 만을 표현
 - 한 다이어그램에 속해 있는 여러 요소 중 특정한 요소의 크기를 다른 요소에 비해 크게 그리는 것은 해당 요소를 중요하게 생각하는 오해를 불러 올 수 있으므로 각 요소의 크기를 통일
 - Note 활용으로 가시적 효과를 이용하여 중요한 특성을 부각

유스케이스 모델링(Use case Modeling)

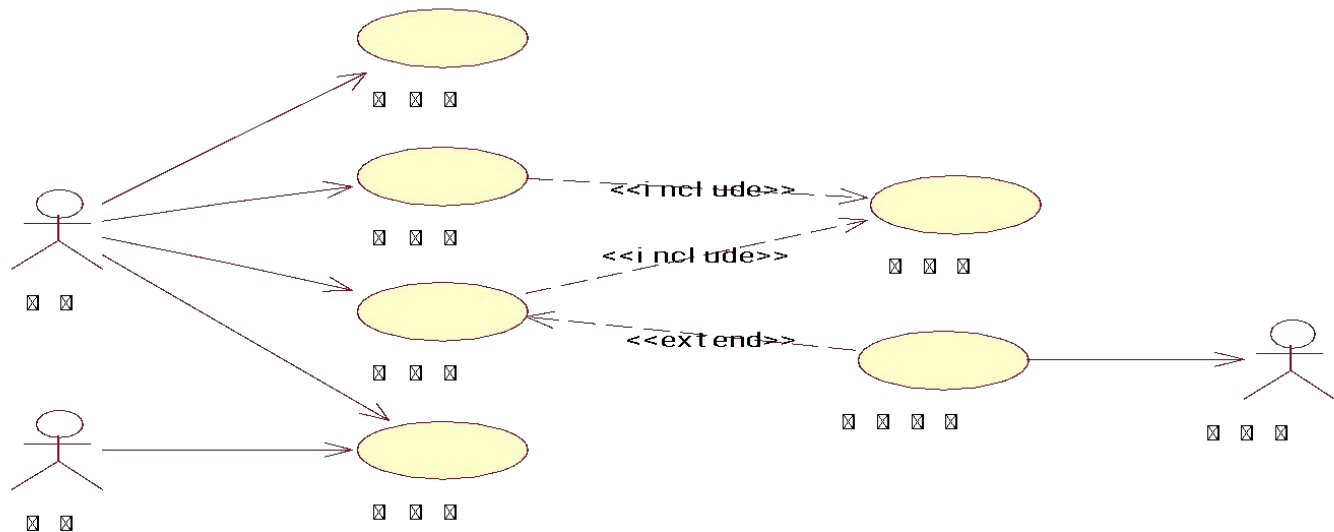
6) Usecase Diagram 작성 (2/2)

- Usecase Diagram 작성지침 (2/2)

- 배치

- » Actor와 UseCase는 중요한 순서에 따라 top-to-bottom, left-to-right순으로 정렬
 - » 행동, 역할, 의미가 연관이 있는 것은 인접하여 배치
 - » 교차되는 Line이 가능하면 최소화 되도록 요소를 배치
 - » Included UseCase는 invoking UseCase의 오른쪽에 위치
 - » Extending UseCase는 parent UseCase의 아래에 위치

- Sample



유스케이스 모델링(Use case Modeling)

- 실습
 - Problem Statement

우리는 호텔의 Internet 예약시스템을 만들고자 한다.

이 시스템에서, 고객은 객실유형으로 객실을 예약한다.

예약 시 고객은 반드시 고객의 정보를 등록하여야 하고, 빠른 업무처리를 위하여 고객의 세부정보는 저장하도록 하여야 하며, 예약이 정상적으로 처리되면 예약번호를 고객에게 알린다.

고객은 객실예약을 취소하고자 하는 경우, 예약번호를 입력하여 객실예약을 취소한다.

호텔에는 예약담당자가 객실예약에 관한 사항을 담당하고, 객실예약 후 투숙하지 않는 건을 처리한다.

시스템은 호텔의 Check In, Check Out에 관한 업무를 처리하여야 한다.

Check In 시에 고객은 예약번호를 입력하고 객실을 배정 받는다.

Check Out 시에 고객은 객실번호를 입력하고 계산서를 발급 받는다.

직원은 고객이 투숙한 계산서에 의해 숙박비를 수납하며, 고객은 현금,

신용카드, 수표를 이용하여 숙박비를 지불 할 수 있다. 수납이 정상적으로

유스케이스 모델링(Use case Modeling)

2. Usecase 명세서(Description)

1) Usecase Description ?

- 시스템의 기능에 대한 세부적인 서술 표현
- 개요와 이벤트 흐름으로 구성
- Usecase들이 문서화되면서, 다른 Usecase 관계들이 발견될 수 있음 : Include, Extend

2) 구성요소 (1/3)

- 이름 (Name)
- Actor 기술 (Description of Actor)
 - 요약 (Summary, Brief Description), 역할 (Role)
- Usecase 기술 (Description of Usecases)
 - 요약 (Summary, Brief Description)
 - 속성 (Attributes)
 - 사전조건 (Pre-condition)/사후조건 (Post-condition)
 - 주 사건 흐름 (Main flow of event)
 - 대안 흐름 (Alternative flow)
 - 예외 흐름 (Exception flow)
 - 시나리오 (Scenario)
 - 기타
 - » 비기능적 요구사항(Non-Functional Requirements)
 - » 비즈니스 룰 (Business Rule)
 - » 사용자 화면 (User Interface),...

유스케이스 모델링(Use case Modeling)

2) 구성요소 (2/3)

- 개요
 - 몇 줄 정도로 Usecase의 목적을 기술
- 이벤트 흐름
 - Usecase가 초기화 되어 발생한 사건들의 기본흐름과 대안흐름
 - » 기본 흐름
 - . 기본 흐름은 모든 것이 제대로 작동하는 것처럼 작성한다 : 해피데이 시나리오
 - . 기본 흐름은 분기나 대안 흐름이 없는 단순한 형태의 문장들이다.
 - » 대안 흐름
 - . 대안 흐름은 기본 경로에 대한 대안을 제시해준다.
 - . 예러도 대안 흐름으로 문서화할 수 있다.
 - . 대안 흐름은 언제든지 일어날 수 있는 일을 보여주는데 도움이 된다.
 - . 정상적인 이벤트의 흐름을 방해하는 것

유스케이스 모델링(Use case Modeling)

2) 구성요소 (3/3)

- 시나리오

- Usecase 내부의 흐름을 바탕으로 실제 상황을 연출하는 것
- Actor와 시스템 사이에서 발생하는 상호작용과 행동들의 특정 절차로서 Usecase의 특정한 예를 나타내는 instance
- Usecase를 통해서 하나의 특정한 경로를 선택하는 것
- 모든 시나리오를 통합하면 완성된 Usecase를 얻게 된다.
- 시나리오는 텍스트로 문서화하거나, 상호작용에 관한 다이어그램을 통해 표현

- 사용자 인터페이스

- 샘플 UI는 스토리보드 형식으로 나타낼 수 있다
 - » 심각한 문제를 발견하거나 새로운 요구사항을 발견 가능
 - » 텍스트에서는 분명하지 않았던 대안들을 발견하는데 도움
- UI를 설계하기 위해 도구를 사용하는 것은 사용자가 애플리케이션에 나타나는 화면에 대한 의견을 제시하는데 도움이 된다

- Usecase 와 Event Flow

- Usecase는 System이 무엇을 해야 하는가를 설명하며, 어떻게 하는가는 명세화 하지 않음
- System의 내부 관점과 외부 관점 간의 관심사를 명백히 분리 표현
- Usecase행동을 명세화 하려면 Event Flow(사건흐름)을 문장으로 설명
- 사건 흐름 작성시 Usecase가 언제, 어떻게 시작/종료 되는지를 표현
- 언제 Actor와 교류하며, 변경 객체와 기본 행동 흐름, 대안 행동 흐름도 표현

- Usecase 와 Scenario

- Scenario는 기본적인 Usecase의 한 Instance임
- 하나의 Usecase는 하나의 Sequence가 아닌 여러 개의 Sequence를 하꺼번에 설명하므로 이러한

유스케이스 모델링(Use case Modeling)

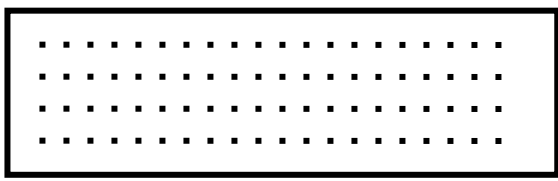
3) Usecase Description 작성 (1/3)

- Usecase Description 작성을 위한 가이드라인
 - Usecase는 Actor의 관점에서 작성한다.
 - » Usecase 내의 모든 단계는 Actor에게 가시적이거나 Actor가 추측할 수 있어야 한다.
 - Usecase를 읽을 사람을 고려한다.
 - » 시스템 사용자, 업무 전문가, 관리자, 개발자, ...
 - 너무 세부적이지 않도록 한다.
 - » 이 시점에서는 요구사항을 수집하는 것이지 세부적인 분석이나 설계를 하는 것이 아니다.
 - 완벽한 Usecase를 만들려고 노력할 필요는 없다.
 - 기타 특별 요구사항에 대한 기술
 - » Actor에게 가시적이지 않거나 Usecase 내에 표현하기 어려운 요구사항
 - . 비기능적 요구사항
 - . 실시간 시스템에 대한 타이밍과 사이즈 요구사항
 - » Usecase 설명서에 특별 요구사항을 추가할 수 있다.
 - . 일반적으로 특별 요구사항들은 비기능적 요소들이 많으며, 이들은 아키텍처 측면에서 중요하게 취급을 할 필요가 있다.

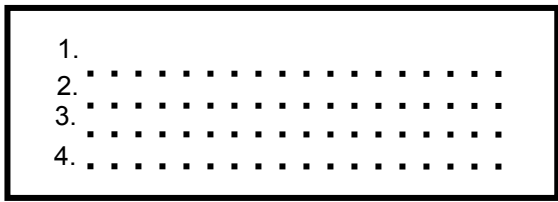
유스케이스 모델링(Use case Modeling)

3) Usecase Description 작성 (2/3)

- 문서 스타일
 - 특정한 형식이 없는 방식



- 숫자를 사용해서 순서를 표현하는 형식



- 테이블 형식

Actor	System
1.	
2.	
	3.
	4.

3) Usecase Description 작성 (3/3)

- Sample

Use Case Description

1. 개요

국방조달 전자상거래체계 확대사업의 기능을 상세하게 기술한다

2. DESCRIPTION OF ACTORS

2.1.원기관리자

- 개요
조달본부에 근무하며, 원기관리를 담당하는 사람이다.
- 역할
원가거래에 대해 서류를 심사하며, 심사한 내용을 등록한다.

3. DESCRIPTION OF USE CASE

3.1.USCAAA(원기자료 인기관리)

3.1.1. USCAAA001: 원기자료 인기관리내역조회

- 개요
원가자료를 의뢰한 내역을 조회한다.
- Initiator
원기관리자
- Attributes
조회일자, 업체구분, 업체명, 사업자등록번호, 주소, 전화번호
- 전제조건
원기관리자로 Log-In 이 되어있어야 한다.

□ 업무 흐름

구분	Aotor	System
주요흐름	1. 원가자료 의뢰내역조회를 요청한다. 3. 조회일자를 입력하고 업체구분을 선택한다.	2. 조회일자를 입력하고 업체구분을 선택할 수 있는 화면을 출력한다. 4. 조회일자와 업체구분에 해당하는 업체를 출력한다.
대체흐름	3. 업체구분을 상세하게 입력하고자 하는 경우 3-1. 업체구분상세입력을 요청한다. 3-3. 업체구분을 선택한다.	3-2. 업체구분이 상세하게 된 테이블을 출력한다.
예외상황		4. 조회일자가 검색일자보다 더 빠른 경우에는 오류메시지를 출력한다.

□ 시나리오

- 홍길동은 원가자료 의뢰내역조회를 요청한다.
- 시스템은 조회일자를 입력하고 업체구분을 선택할 수 있는 화면을 출력한다.
- 홍길동은 조회일자에 "20020115"를 입력하고, 업체구분에서 "대기업"을 선택한다.
- 시스템은 조회일자와 업체구분에 해당하는 업체명 "현대정보기술", 업체사업자번호"111-111-11111", 업체주소 "서울특별시 중구 충무로 3 가 60-1", 업체전화번호"02-999-9999"를 출력한다.

유스케이스 모델링(Use case Modeling)

4) Check Points

- Usecase가 정상적으로 수행되기 위하여 가정하고 있는 상황을 선행조건을 통하여 파악할 수 있는가?
- Usecase의 수행결과를 후행 조건을 통하여 파악할 수 있는가?
- 선행조건과 후행조건만으로 Usecase의 기능을 개략적으로 이해할 수 있는가?
- 이벤트 흐름을 통하여 Actor가 어떻게 시스템과 상호작용을 하여 Usecase의 기능을 이용하는지 명확하게 파악할 수 있는가?
- 기본 흐름을 포함하여 대안흐름이 충분히 정의 되었는가?
- 정의된 시나리오에 실제 값으로 명확히 기술되어 있으며 테스트 케이스로 활용 가능한가?
- 해당 Usecase의 기능을 만족하기 위한 비기능적 요구사항 및 사용자 인터페이스 등의 특이한 사항이 기록되어 있는가?

유스케이스 모델링(Use case Modeling)

- 실습
 - Usecase Description
 - 예약 Usecase
 - » 유희객실을 조회한 뒤, 객실예약