유스케이스 다이어그램

시스템과 사용자의 상호작용을 다이어그램으로 표현한 것으로 사용자의 관점에서 시스템의 서비스 혹은 기능 및 그와 관련한 외부 요소를 보여주는 것이다.

사용자가 시스템 내부에 있는 기능 중에 어떤 기능을 사용 할 수 있는지 나타내며 유스케이스 다이어그램을 사용함으로써 고객과 개발자가 요구사항에 대한 의견을 조율 할 수 있다.

한마디로 사용자랑 시스템사이에 관계를 나타내는 것으로 볼 수 있다.

유스케이스 다이어그램은 프로젝트에 대한 요구사항을 정의하고 세부기능을 분석하며 개발 범위를 정할 때 작성한다.

구성요소(Component)

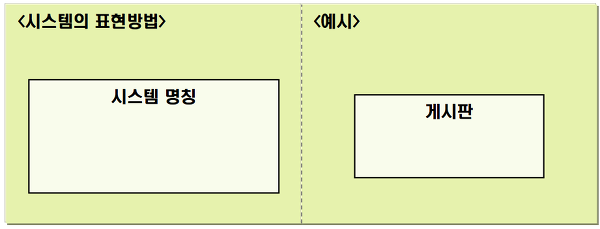
유스케이스 다이어그램의 구성요소는 시스템(System), 액터(Actor), 유스케이스(Usecase), 관계(Relation)로 구성되어 있다.

1) 시스템(System)

만들고자 하는 프로그램을 나타낸다.

-표기

유스케이스들을 둘러싼 사각형 틀로 시스템 명칭을 안쪽 상단에 작성한다.



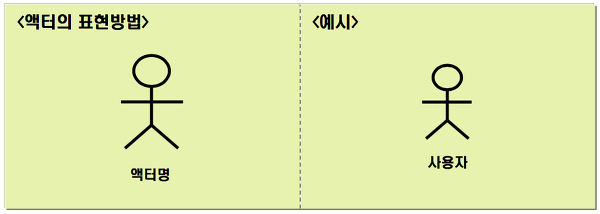
2) 액터(Actor)

시스템의 외부에 있고 시스템과 상호작용을 하는 사람(시스템의 기능을 사용하는 사람), 시스템(시스템에 정보를 제공하는 또 다른 시스템)을 말한다.

-표기

원과 선을 조합하여 사람~~(졸라맨)~~ 모양으로 표현

액터명은 위나 아래에 표시하며 액터의 역할을 작성한다.



3) 유스케이스(Usecase)

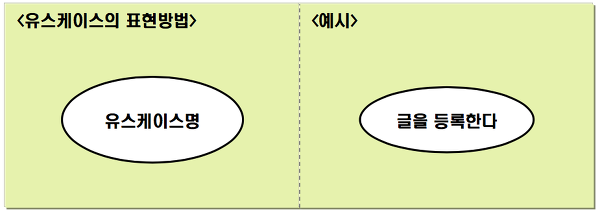
사용자 입장에서 바라본 시스템의 기능

시스템이 액터에게 제공해야 하는 기능으로 시스템의 요구사항을 나타낸다.

-표기

타원으로 표시하고 안쪽에 유스케이스명을 작성한다.

유스케이스명은 "~한다"와 같이 동사로 표현한다.

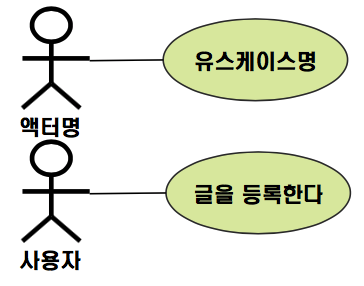


4) 관계(Relation)

액터와 유스케이스 사이의 의미있는 관계를 나타낸다. 종류는 연관(Association), 의존(Dependency), 일반화(Generalization)이 있으며 의존관계는 포함(Include), 확장(Extend)로 나눠진다.

1. 연관관계(Association)는 유스케이스와 액터간의 상호작용이 있음을 표현한다.

유스케이스와 액터를 실선으로 연결한다.

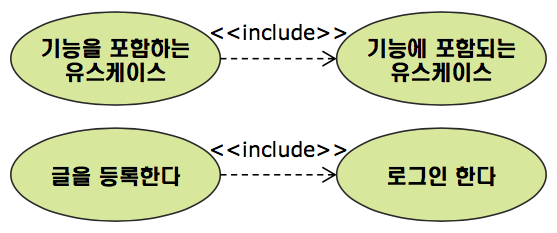


위 그림은 "사용자"(액터)가 "글을 등록한다"(유스케이스)는 기능과 상호작용이 있다는 것을 나타낸다.

2. 포함 관계(Include)는 하나의 유스케이스가 다른 유스케이스의 실행을 전제로 할 때 형성되는 관계이다.

포함되는 유스케이스는 포함하는 유스케이스를 실행하기 위해 반드시 실행되어야 하는 경우에 적용한다.

포함하는 유스케이스에서 포함되는 유스케이스 방향으로 화살표를 점선으로 연결하고 <<include>>라고 표기한다.

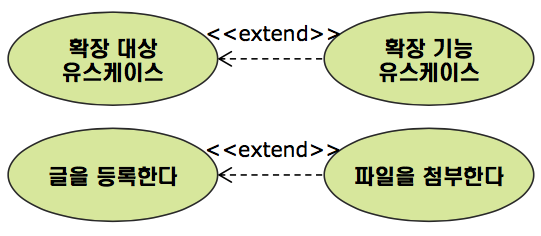


위 그림은 "글을 등록한다" 기능을 동작하기 위해서 "로그인 한다" 기능이 반드시 동작되어야 한다는 것을 나타낸다.

3. 확장 관계(Extend)는 확장 기능 유스케이스와 확장 대상 유스케이스 사이에 형성 되는 관계이다.

확장 대상 유스케이스를 수행 할 때 특정 조건에 따라 확장 기능 유스케이스를 수행하는 경우에 적용한다.

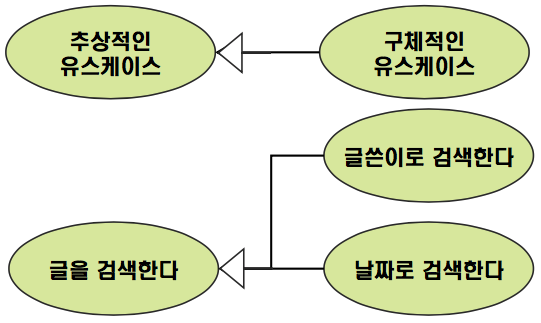
확장 기능 유스케이스에서 확장 대상 유스케이스 방향으로 화살표를 점선으로 연결하고 <<extend>>라고 표기한다.



위 그림은 "글을 등록한다" 기능을 수행 할 때 "파일을 첨부한다" 기능을 선택적으로 수행 할 수 있다는 것을 나타낸다.

4. 일반화 관계(Generalization)는 유사한 유스케이스 또는 액터를 모아 추상화한 유스케이스 또는 액터와 연결시켜 그룹을 만들어 이해도를 높이기 위한 관계이다.

구체적인 유스케이스에서 추상적인 유스케이스 방향으로 끝부분이 삼각형으로 표현된 화살표를 실선으로 연결하여 표현한다.



위 그림은 "글을 검색한다"를 "글쓴이로 검색한다"와 "날짜로 검색한다"로 좀더 구체화 한 것을 나타낸다.

작성 순서

유스케이스 작성 순서로는 1. 액터 식별 2. 유스케이스 식별 3. 관계 정의 이다.

**1. 액터 식별**

액터는 시스템에 관련이 있는 사용자의 역할과 외부 시스템으로 식별 할 수 있다.

액터를 식별하기 위한 질문?

 - 누가 정보를 제공하고, 사용하고, 삭제하는가?

 - 누가 또는 어떤 조직에서 개발될 시스템을 사용할 것인가?

 - 누가 요구사항에 대해 관심을 가지고, 시스템이 만들어낸 결과에 관심이 있는가?

 - 누가 시스템이 잘 운영될 수 있도록 유지보수 및 관리를 하는가?

 - 개발될 시스템과 상호작용하는 하드웨어나 소프트웨어 시스템은 무엇인가?

**2. 유스케이스 식별**

액터가 요구하는 서비스, 정보를 유스케이스로 식별 할 수 있고 액터가 시스템과 상호작용하는 행위를 유스케이스로도 나타낼 수 있다.

유스케이스를 식별하기 위한 질문?

 - 액터가 원하는 시스템 제공 기능은 무엇인가?

 - 액터는 시스템에 어떤 정보를 생성, 수정, 조회, 삭제하고 싶어 하는가?

 - 모든 기능 요구사항들을 만족할 수 있도록 유스케이스가 모두 식별 되었는가?

**3. 관계 정의**

액터간, 유스케이스간의 일반화, 연관관계를 정의하고 유스케이스간의 포함, 확장관계를 정의한다.

관계를 식별하기 위한 질문?

 - 연관 관계: 액터와 유스케이스 간에 상호 작용이 존재하는가?

 - 포함 관계: 유스케이스를 실행하기 위하여 반드시 실행되어야 하는 유스케이스가 존재하는가?

 - 확장 관계: 유스케이스를 실행함으로써 선택적으로 실행되는 유스케이스가 존재하는가?

 - 일반화 관계: 액터 또는 유스케이스가 구체화 된 다른 액터 또는 유스케이스를 가지고 있는가?