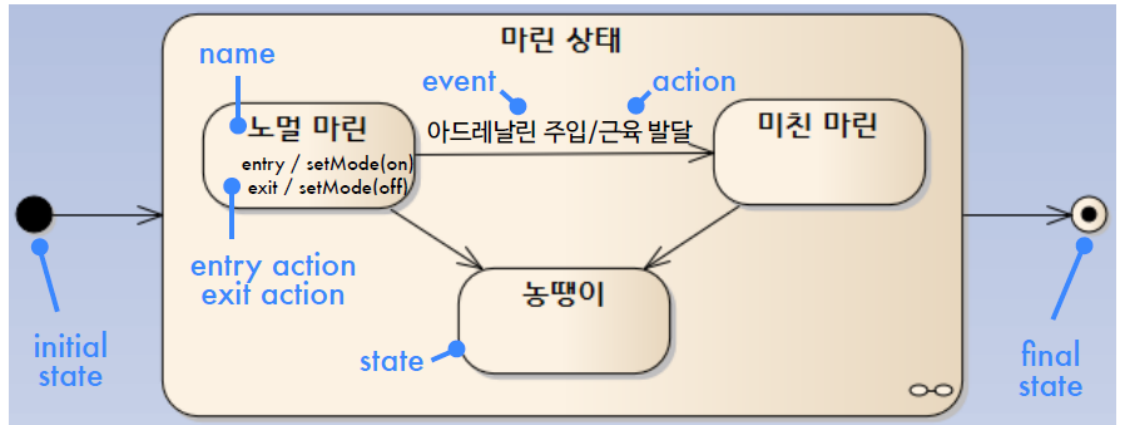


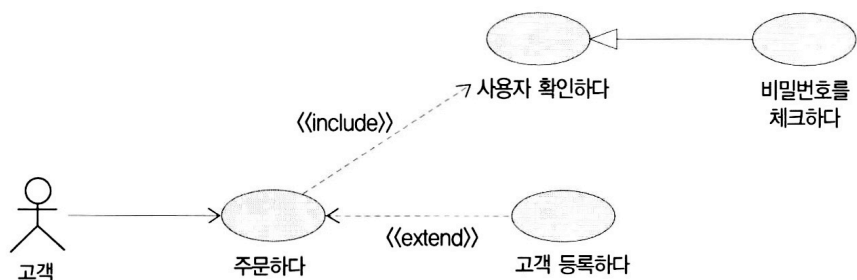
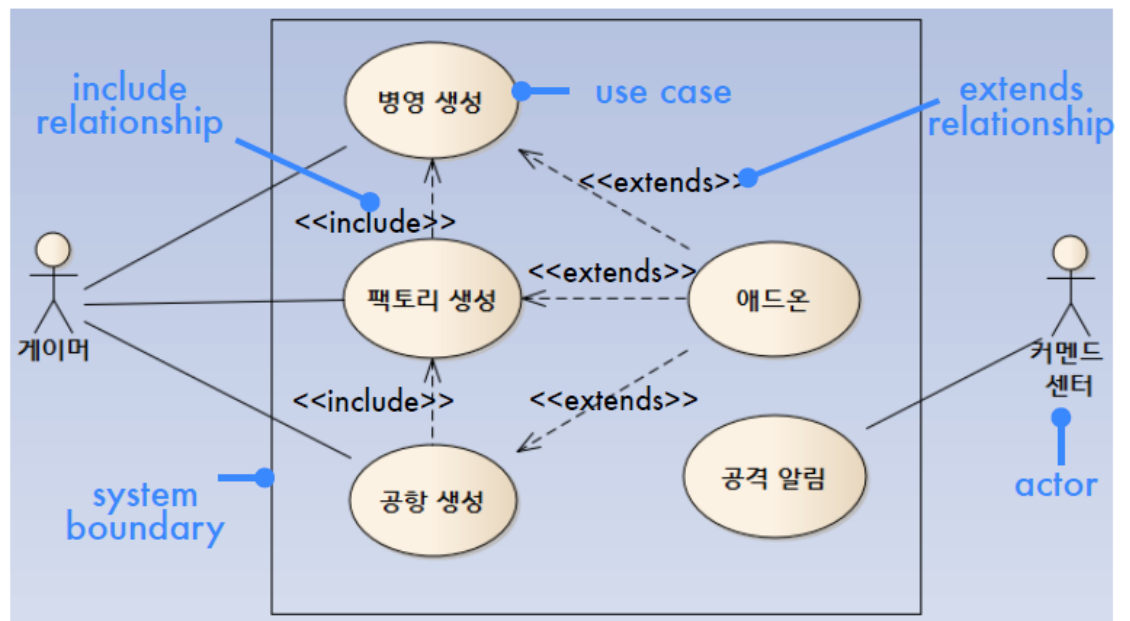
## State Machine Diagram

**[특성]** 클래스의 객체가 가질 수 있는 모든 가능한 상태, 특정 객체에 대한 사건 발생에 따른 상태 천이과정



## Usecase Diagram

**[특성]** 사용자와 상호작용하는 시스템의 모습, **시스템이 제공하는 기능** 설명



## Usecase 작성



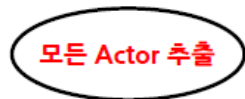
시스템 외부에 존재하면서 시스템과 교류하는 **시스템 또는 사람**  
개발하고자 하는 시스템과 연동되는 모든 외부 시스템(사람 포함)은 액터로 식별되어야 함



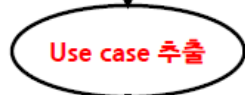
완성된 시스템이 제공하게 될 기능을 상위 수준에서 기술한 **요구사항 단위**  
**액터가 시스템에 요청하는 서비스의 단위**

유즈케이스간의 업무 흐름관계는 없음

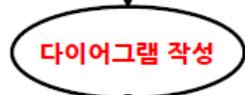
액터가 타 시스템일 경우, 유즈케이스간의 관계 자체가 인터페이스가 됨



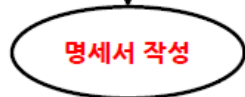
요구사항 명세서를 기준으로 사람, 시스템 등의 이해관계자를  
모두 Actor 후보군으로 설정하고 추출한다.



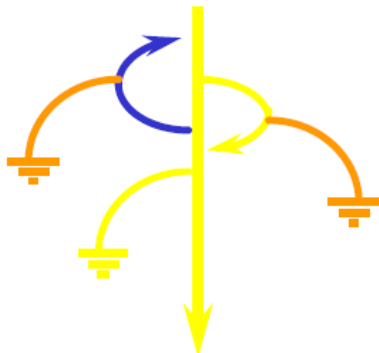
요구사항 명세서에서 Actor간의 연관성을 제시하는  
동작을 기반으로 Use case로 설정하고 추출한다.



추출된 Actor와 Use case를 통해 System Boundary를 설정,  
포함, 확장 관계를 기반으로 다이어그램을 작성한다.



Actor, Use case, System Boundary, 관계를 기반으로  
Basic Flow, Alternative Flow, Exception Flow 및  
선/후행 조건을 기술하는 Use case specification을 작성한다.



### 유즈케이스 명세서란?

유즈케이스의 Flow of Event를 기술한 문서  
액터와 유즈케이스간 정보 교류 기술  
UI에 관한 내용은 언급하지 않음

### Flow of Event

#### Base Flow

- 일반적인 흐름을 기술
- 잘못된 경우는 기술하지 않음

#### Alternative Flow

- Basic Flow에서 오류가 아닌 예외사항을 분리하여 기술

#### Exception Flow

- 오류 발생 후 유즈케이스가 종료되는 조건을 기술

## <<include>> vs <<extend>>

	포함관계	확장관계
목 적	• 여러 유스케이스에 공통적인 기능을 표현하기 위해 사용된다.	• 기존 유스케이스에 부가적으로 추가된 기능을 표현하기 위해 사용된다.
이벤트 흐름	• 포함된 유스케이스로의 이벤트 흐름 분기가 필수적이다. • 기존 유스케이스 이후의 이벤트 흐름이 포함된 유스케이스의 수행 결과에 의존한다.	• 확장 유스케이스에 기술된 조건에 따라 선택적으로 수행된다. • 기존 유스케이스 이후의 이벤트 흐름이 확장 유스케이스의 결과에 의존하지 않는다.

업무명	공정	대분류	XXXX	중분류	XXXX
유스케이스명	Slab 단위 조정	유스케이스 ID	SM-UC-001	관련 액터	공정계획담당자
		작성 일자	2006.11.23	작성 자	홍길동

Basic Flow	Alternative Flow
<ol style="list-style-type: none"> <li>1. 공정 계획 담당자가 Roll 배치 정보를 요청 한다.</li> <li>2. 시스템은 Roll 단위 배치 정보 정보를 저장소에서 조회한다.</li> <li>3. 시스템은 조회된 결과를 출력한다.</li> <li>4. 공정 계획 담당자는 결과 화면에서 설계 조정할 Heat수를 선택한다.(AF1)</li> <li>5. 시스템은 선택된 Roll 정보 및 Cast 정보를 이용하여 Heat 설계 정보 및 Slab 정보를 조회한다.</li> <li>6. 시스템은 조회된 결과를 임시 저장소에 저장한다.</li> <li>7. 시스템은 조회된 결과를 화면에 출력한다.</li> <li>8. 공정 계획 담당자는 조정할 Slab를 이동한다.</li> <li>9. 시스템은 Slab 조정이 가능한지 체크 한다.(AF2)</li> <li>10. 시스템은 이동 결과를 임시 저장소에 저장한다.</li> <li>11. 시스템은 각 Heat별 총 중량을 재 계산한다.</li> <li>12. 시스템은 수정된 정보를 Roll 단위 배치 정보, Heat 및 Cast 설계 정보가 저장소에 저장한다.</li> <li>13. 시스템은 Roll 단위 배치 정보 정보를 저장소에서 조회한다.</li> <li>14. 시스템은 조회된 결과를 출력한다.</li> <li>15. Use Case 종료.</li> </ol>	<p>AF1 Roll의 순서를 변경할 경우</p> <ol style="list-style-type: none"> <li>1. 공정 계획 담당자가 순서 조정할 Roll을 선택 후 위치를 이동한다.</li> <li>2. 시스템은 변경된 정보를 Roll 단위 배치 정보 저장소에 저장한다.</li> <li>3. 시스템은 Roll 단위 배치 정보를 저장소에서 조회한다.</li> <li>4. 시스템은 조회된 결과를 출력한다.</li> <li>5. Use Case 종료</li> </ol> <p>AF2 순서 조정 불가일 경우</p> <ol style="list-style-type: none"> <li>1. 시스템은 조정 불가 메시지를 출력한다.</li> <li>2. Use Case 종료</li> </ol>

### Usecase 에서 클래스 도출

구분	설명
Entity Class 엔티티 클래스	시스템에서 영구적으로 저장되어 사용될 자료를 보관하는 클래스
Boundary Class 경계 클래스	주로 시스템 외부의 액터와 상호 작용하는 클래스
Control Class 제어 클래스	Boundary Class와 Entity Class 사이에 중간 역할을 하는 클래스 비즈니스 로직을 담고 있는 클래스 자료를 다른 클래스로부터 받아 처리하는 것이 주 임무인 클래스

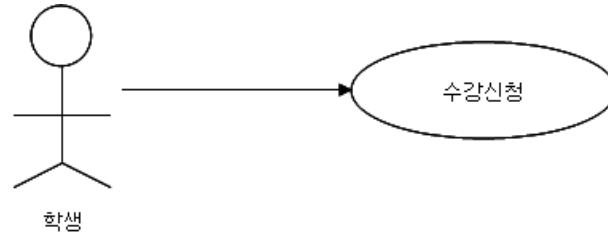
### [도출 예시]

요구사항

- 학생은 개설된 강의에 대하여 수강신청을 할 수 있음
- 개설된 강의는 최대 수강인원이 정해지면 이 인원을 초과할 수 없음
- 학생은 18 학점까지만 신청할 수 있고, 동일강좌의 중복 수강은 허용되지 않음
- 학생은 수강할 강의를 등록하거나 기존에 수강 신청된 강의를 취소할 수 있음

### 유즈케이스 다이어그램 작성

- 1) 액터식별 : 학생
- 2) 유스케이스 식별 : 수강신청
- 3) 유스케이스 다이어그램



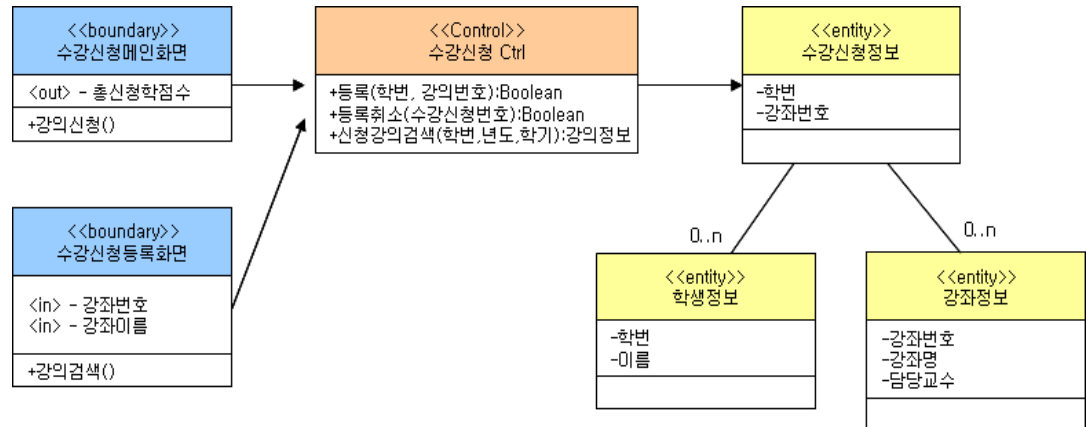
### 유즈케이스 명세서 작성

개요	학생은 수강할 강의를 신청하거나 변경한다
관련액터	학생
우선순위	상
선행조건	학생은 로그인 되어 있어야 한다.
이벤트 흐름	<p>기본흐름</p> <ol style="list-style-type: none"> <li>1. 학생은 학생 메인화면에서 수강신청을 선택한다.</li> <li>2. 시스템은 수강신청 메인화면을 보여준다. 수강신청 메인화면에서 현재 신청된 강의 목록과 신청된 총 학점수를 보여준다. 그리고 신청된 강의를 취소할 수 있게 제거 버튼을 보여준다.</li> <li>3. 학생은 수강신청 메인화면에서 개설된 강의를 수강 신청하거나 신청된 강의를 취소한다.</li> </ol> <p>등록</p> <ol style="list-style-type: none"> <li>1. 학생은 수강신청 메인화면에서 수강신청을 선택한다.</li> <li>2. 시스템은 수강신청 등록화면을 보여준다.</li> <li>3. 학생은 강좌번호와 강좌이름으로 개설된 강의를 검색한다.</li> <li>4. 시스템은 주어진 검색조건에 일치하는 강의목록을 구하여 화면에 표시한다.</li> <li>5. 학생은 신청할 강의를 선택한다.</li> <li>6. 시스템은 선택된 강의를 학생이 신청한 것으로 기록한다.</li> <li>7. 시스템은 학생이 신청한 강의를 확인할 수 있게 수강신청 메인화면을 보여준다.</li> </ol> <p>삭제</p> <ol style="list-style-type: none"> <li>1. 학생은 수강신청 메인화면에서 수강신청을 취소할 강의의 제거 버튼을 누른다.</li> <li>2. 시스템은 해당 수강신청에 대한 정보를 삭제하고 수강신청 메인화면을 갱신한다.</li> </ol>
후행조건	요청된 수강신청에 대한 등록/삭제가 이루어진다.

### 클래스 도출

- 1) 바운더리 클래스: 수강신청 메인화면, 수강신청 등록화면

- 2) 컨트롤 클래스: 수강신청 Ctrl  
3) 엔티티 클래스: 수강신청정보, 학생정보, 강좌정보



[특성] 객체들 사이의 동적인 협력사항, 시간과 순서에 따라 묘사

Sequence Diagram

