

# OSP stage 2030

---

## Intelligent Elevator Controller

---

### 3조

200312468

김완수

200511363

한상현

200960122

사인빌릭 체렝밤바

# Activity 2031. Define Essential Use Cases

## 1. Open Door

<b>UseCase Name</b>	OpenDoor
<b>Description</b>	- 엘리베이터가 정지가 된 상태에서 사용자의 문열림 버튼이 입력이 있을 때 엘리베이터 하드웨어 문 열림 기능을 전달하여 문이 열리게 한다. - 사용자의 문열림 요청이 없더라도 버튼으로 입력된 목적지에 도착 했을시에 문이 열리도록 하드웨어에 전달 한다.
<b>Actors</b>	None
<b>Type</b>	Primary
<b>Cross Reference</b>	System functions R1.1.1, R1.1.2, R1.1.3, R2.1.1 UseCase : CloseDoor, MoveEle, InputButton
<b>Preconditions</b>	None
<b>Typical Course of Event</b>	(A) : Actor, (S) : System 1. (A) 사용자는 외부 버튼 박스의 화살표 키를 누르거나 엘리베이터 내부의 버튼 박스의 층버튼 또는 문열림 버튼을 입력. 2. (S) 엘리베이터가 운행중이지 않을 경우 MoveEle를 통하여 내·외부의 입력 받은 층으로 이동. 3. (S) 입력받은 층에 도착했을시에 문을 엮. 4. (S) 처리한 층에 대한 입력버튼 혹은 화살표버튼의 입력을 없앴. 5. (S) 일정 시간 문이 열린 시간이 지나면 문을 닫는다.
<b>Alternative Courses of Events</b>	None
<b>Exceptional Courses of Events</b>	Line5. 사용자가 문열림 버튼 또는 외부의 화살표 버튼을 입력했을시에는 문이 열린시간을 재는 타이머를 초기화 시킨다.

# Activity 2031. Define Essential Use Cases

## 2. Close Door

<b>UseCase Name</b>	CloseDoor
<b>Description</b>	<ul style="list-style-type: none"> <li>- 문이 열려있을 때에 사용자가 문닫힘 버튼을 입력하였을 때 문을 닫는 명령을 엘리베이터 하드웨어에 전달한다.</li> <li>- 사용자의 문닫힘 요청이 없더라도 문이 열린 후 지정된 시간 이후에 자동으로 문이 닫히게 한다.</li> <li>- 문이 닫힐때에는 물체감지센서를 이용하여 문 사이에 어떠한 물체가 없는지 판단하고 발견시에 문열림 기능을 실행하게 한다.</li> <li>- 통계에 기록되거나 사용자의 인식 또는 사라짐을 인식하는 시점은 문이 닫히는 시점으로 하기 때문에 통계입력 및 사용자 인식 등의 모든 데이터 베이스 처리를 한다.</li> <li>- 자동대기 기능을 실행 할 것인가 하는 기능도 문이 닫혔을 때 사용자 및 엘리베이터의 이동을 감지하여 실행하게 한다.</li> </ul>
<b>Actors</b>	None
<b>Type</b>	Primary
<b>Cross Reference</b>	System functions R1.1.1, R1.1.2, R2.1.1, R3.1, R3.2, R5.1, R5.3 UseCase : OpenDoor, CloseDoor, InputButton, WeightConfirm SafeDoor, SearchGuest, DatabaseInput
<b>Preconditions</b>	- 문이 열려있는 상태에서만 동작한다.
<b>Typical Course of Event</b>	<p>(A) : Actor, (S) : System</p> <ol style="list-style-type: none"> <li>1.1. (A) 문닫힘 버튼을 선택한다.</li> <li>1.2. (S) 문이 열려져있는 상태가 지속됨을 인지한 UseCase1로부터 명령을 받는다.</li> <li>2. (S) 문을 닫기전 엘리베이터에서 수용할 수 있는 무게가 엘리베이터에 탔는지 점검후 허용되지 않으면 문을 닫지 않는다.</li> <li>3. (S) 문이 완전히 닫히기 전까지 문사이의 사물파악센서를 통하여 중간에 끼거나 또다른 사용자가 접근 중인지에 대하여 파악.</li> <li>4. (S) 해당 층수 버튼의 입력으로부터 엘리베이터의 이동시간이 얼마만큼 오래 걸렸는지 파악.</li> <li>5. (S) 카메라를 통하여 이전에 타고 있던 사용자 포인터가 사라진 수를 체크.</li> <li>6. (S) 카메라를 통하여 새로 탑승한 사용자의 수를 파악.</li> <li>7. (S) 4~6번의 데이터를 DB에 입력.</li> </ol>
<b>Alternative Courses of Events</b>	None
<b>Exceptional Courses of Events</b>	Line1.1 문 닫힘 버튼과 문 열림 버튼의 동시 입력 시에는 문 열림 버튼에 우선권을 부여한다.

# Activity 2031. Define Essential Use Cases

## 3. MoveEle

<b>UseCase Name</b>	3. MoveEle
<b>Description</b>	<ul style="list-style-type: none"><li>- 엘리베이터가 위로 혹은 아래로 움직이는 기능. 입력받은 층과 현재 엘리베이터 위치층을 비교해서 입력받은 층이 크면 위로, 작으면 아래로 이동하고 해당된 층까지 이동을 하게 되면 자동정지하고 문을 개방한다.</li><li>- 문 닫힘 상태일때만 수행이 가능하다.</li><li>- 사용자로부터 입력을 받아서 수행하는 것을 기본으로 하되 자동대기기능 실행 시에는 컨트롤러가 주는 신호만으로도 수행이 가능토록 한다</li></ul>
<b>Actors</b>	None
<b>Type</b>	Primary
<b>Cross Reference</b>	System functions R1.1.3, R2.1.1, R3.3, R3.4, R3.5, R3.6 UseCase : OpenDoor, InputButton, DisplayTimer, LocationDisplay, InformDanger, AutoWait
<b>Preconditions</b>	- 문이 닫혀 있을때에만 가능하다.
<b>Typical Course of Event</b>	(A) : Actor, (S) : System 1.1. (A) 사용자가 목적지 층버튼 입력. 1.2. (S) 사용자들의 입력이 전무하고 엘리베이터 내부에 사람이 없을 때 자동 대기 할 층의 정보를 입력. 2. (S) 문이 닫혀 있는지 점검한 후 엘리베이터를 이동. 3. (S) 이동중에 외부대기자의 대기시간을 업데이트 4. (S) 위치센서를 통한 현재의 위치 업데이트 5. (S) 이동중에 InformDanger를 실행
<b>Alternative Courses of Events</b>	None
<b>Exceptional Courses of Events</b>	- 이동시에 문열림 혹은 문닫힘 버튼의 입력은 무효화한다.

# Activity 2031. Define Essential Use Cases

## 4. InputButton

<b>UseCase Name</b>	4. InputButton
<b>Description</b>	<ul style="list-style-type: none"> <li>- 엘리베이터 내부/외부를 포함한 모든 버튼의 입력을 받아들이는 기능</li> <li>- 외부 버튼에는 위/아래로 이동예약을 하는 2개의 버튼, 내부 버튼에는 층수입력 / 문열림 / 문닫힘 / 관리자호출 기능을 하는 버튼이 있으며 입력이 들어왔을 시 하드웨어 장비에게 신호를 보낸다.</li> <li>- 버튼박스의 개수가 n개일 경우 하나의 입력을 모든 버튼박스에 디스플레이 해주는 멀티 인풋기능을 실행 한다.</li> </ul>
<b>Actors</b>	Passenger
<b>Type</b>	Primary
<b>Cross Reference</b>	System functions R1.1.1, R1.1.2, R1.1.3, R2.1.1, R2.1.2, R2.2, R2.3 ,R3.5 UseCase : OpenDoor, CloseDoor, MoveEle, InputButton, CancelButton, Call Admin, MultiInput, DisplayTimer
<b>Preconditions</b>	None
<b>Typical Course of Event</b>	<p>(A) : Actor, (S) : System</p> <ul style="list-style-type: none"> <li>* (A) 버튼을 입력.</li> <li>-&gt; (S) 버튼박스가 여러개일 경우 멀티인풋기능을 실행.</li> <li>* (A) 원하는 층수버튼 또는 화살표 버튼을 입력.</li> <li>-&gt; (S) 해당 층으로 이동.</li> <li>* (A) 문열림 또는 문닫힘 버튼을 입력.</li> <li>-&gt; (S) 문열림 또는 문닫힘 기능을 실행.</li> <li>* (A) 관리자 호출 버튼을 입력.</li> <li>-&gt; (S) 관리자호출 기능을 실행.</li> <li>* (A) 외부 화살표 버튼 입력</li> <li>-&gt; (S) 대기예상시간 표시</li> </ul>
<b>Alternative Courses of Events</b>	None
<b>Exceptional Courses of Events</b>	- 현재 엘리베이터가 위치한 층을 입력 했을 시에는 버튼 입력을 취소한다.

# Activity 2031. Define Essential Use Cases

## 5. CancelButton

<b>UseCase Name</b>	5. CancelButton
<b>Description</b>	<ul style="list-style-type: none"><li>- 입력받은 버튼의 기능을 취소하는 기능</li><li>- 처음 입력받은 버튼의 내용이 실행완료 되기 전 같은 입력이 들어온다면 그에 해당하는 실행내용을 취소한다.</li><li>- 취소버튼도 마찬가지로 버튼박스의 개수가 n개일 경우 하나의 취소 입력을 모든 버튼박스에 적용하도록 한다.</li></ul>
<b>Actors</b>	Passenger
<b>Type</b>	Primary
<b>Cross Reference</b>	System functions R1.1.3, R2.1.1, R2.1.2, R2.3 UseCase : MoveEle, MultiInput, InputButton
<b>Preconditions</b>	- 버튼이 현재 입력되어있다.
<b>Typical Course of Event</b>	(A) : Actor, (S) : System 1. (A) 현재 입력된 층수를 취소. 2. (S) 버튼박스의 입력내용을 취소. 3. (S) 모든 버튼박스의 내용을 수정. 4. (S) 현재 엘리베이터의 진행방향을 판단하여 운행, 정지, 역방향 운행 등을 실행.
<b>Alternative Courses of Events</b>	None
<b>Exceptional Courses of Events</b>	Line1. 관리자 호출, 문닫힘, 문열림의 경우 취소가 없다.

# Activity 2031. Define Essential Use Cases

## 6. CallAdmin

<b>UseCase Name</b>	6. CallAdmin
<b>Description</b>	- InputButton에서 관리자호출버튼을 눌렀을시에 제공되는 기능 - 입력이 들어왔을 때 시스템에서 관리자로 설정된 사람의 연락처로 연락을 하거나 인터폰등을 통한 대화를 할 수 있다.
<b>Actors</b>	Passenger
<b>Type</b>	Primary
<b>Cross Reference</b>	System functions R2.1.1, R2.2 UseCase : InputButton, CallAdmin
<b>Preconditions</b>	None
<b>Typical Course of Event</b>	(A) : Actor, (S) : System 1. (A) 사용자가 관리자 버튼을 입력. 2. (S) 인풋박스에서 관리자 호출 기능을 실행. 3. (S) 관리자가 설정한 방법으로 호출을 실시.
<b>Alternative Courses of Events</b>	None
<b>Exceptional Courses of Events</b>	None

# Activity 2031. Define Essential Use Cases

## 7. MultiInput

<b>UseCase Name</b>	7. MultiInput
<b>Description</b>	- 인풋이 여러곳에 있을 때 각기 인풋을 모든 곳에 적용 시킬수 있는 기능 - 버튼박스가 엘리베이터 내부 안에 2개 이상 존재할 때 한쪽에서 받은 입력에 대한 요구를 다른 쪽의 버튼 박스에도 추가할 수 있게 한다.
<b>Actors</b>	None
<b>Type</b>	Primary
<b>Cross Reference</b>	System functions R2.1.1, R2.1.2, R2.3 UseCase : InputButton, CancelButton, MultiInput
<b>Preconditions</b>	- 버튼박스가 여러개 존재한다.
<b>Typical Course of Event</b>	(A) : Actor, (S) : System 1. (A) 사용자가 버튼을 입력. 2. (S) 여러개의 버튼박스에 한쪽에서 입력한 모든 입력값을 표시.
<b>Alternative Courses of Events</b>	None
<b>Exceptional Courses of Events</b>	None

# Activity 2031. Define Essential Use Cases

## 8. WeightConfirm

<b>UseCase Name</b>	8. WeightConfirm
<b>Description</b>	- 엘리베이터 하드웨어 장비의 최대 허용무게를 판별하기 위하여 모터전류의 과부하 값을 가져와서 허용무게를 넘었을시 운행 중단 및 경고음이 울리도록 한다.
<b>Actors</b>	None
<b>Type</b>	Primary
<b>Cross Reference</b>	System functions R1.1.2, R3.1 UseCase : CloseDoor, WeightConfirm
<b>Preconditions</b>	- 엘리베이터의 문이 열려있으며 이제 시간이 되거나 버튼의 입력으로 문이 닫히기 직전이다.
<b>Typical Course of Event</b>	(A) : Actor, (S) : System 1. (A) 사용자의 문닫힘 버튼 입력/ 자동 문닫힘 기능 수행 2. (S) 모터의 전류 측정 센서로부터 엘리베이터 가동 모터의 전류 과부하 값을 가져옴. 3. (S) 모터의 과부하 치가 엘리베이터 하드웨어의 적재량을 초과. 4. (S) 경고음을 울리며 문이 닫히지 않게 한다.
<b>Alternative Courses of Events</b>	None
<b>Exceptional Courses of Events</b>	None

# Activity 2031. Define Essential Use Cases

## 9. SafeDoor

<b>UseCase Name</b>	9. SafeDoor
<b>Description</b>	- 양쪽 문의 감지센서를 통하여 문이 닫힐 때에 센서에 사람 또는 사물이 감지되었을 경우 자동으로 문열림 기능을 실행한다.
<b>Actors</b>	None
<b>Type</b>	Primary
<b>Cross Reference</b>	System functions R1.1.1, R1.1.2, R3.2 UseCase : OpenDoor, CloseDoor, SafeDoor
<b>Preconditions</b>	- 엘리베이터의 문이 열려있다.
<b>Typical Course of Event</b>	(A) : Actor, (S) : System 1. (A) 사용자의 문닫힘 버튼 입력/ 자동 문닫힘 기능 수행 2. (S) 문에 달린 감지센서의 값을 가져옴. 3. (S) 사람이나 물체가 감지되었을 경우 문을 다시 열게함.
<b>Alternative Courses of Events</b>	None
<b>Exceptional Courses of Events</b>	Line1. 문닫힘 버튼이 입력되어도 센서가 사물을 감지하면 무시한다.

# Activity 2031. Define Essential Use Cases

## 10. AutoWait

<b>UseCase Name</b>	10. AutoWait
<b>Description</b>	- 데이터베이스를 통하여 통계가 나온 데이터를 이용하여 특정 알고리즘을 적용 하여 해당범위 내에 해당시간에 대한 데이터의 근사치와 현재의 엘리베이터 위치 등을 고려하여 자동으로 엘리베이터를 자주 이용하는 층에 자주 이용하는 시간대에 대기 시켜준다.
<b>Actors</b>	None
<b>Type</b>	Secondary
<b>Cross Reference</b>	System functions R1.1.3, R5.1, R5.4 UseCase : MoveEle, SeachGuest, DatabaseCalc, SetCamDb
<b>Preconditions</b>	- 엘리베이터 버튼박스의 입력이 전무하고 승객이 현재 엘리베이터 내에 존재 하지 않는다.
<b>Typical Course of Event</b>	(A) : Actor, (S) : System 1. (A) 버튼 입력이 없다. 2. (S) 카메라를 통하여 사용자 유무를 찾음. 3. (S) 현재의 시간값을 가져옴. 4. (S) DB에서 현재의 시간에 대응하는 통계값을 가져옴. 5. (S) 현재의 층수정보를 가져옴. 6. (S) DB에서 현 시간대에 가장 많이 운행한 층 구함 7. (S) 최대층수를 구함. 8. (S) 설정된 자동대기의 설정치에 따라 현재의 층수, 통계된 층수, 최대층수를 종합적으로 계산하여 가야할 층의 인식 및 실행
<b>Alternative Courses of Events</b>	None
<b>Exceptional Courses of Events</b>	- 관리자의 설정에 따라 사용하지 않을수 있다.

# Activity 2031. Define Essential Use Cases

## 11. InformDanger

<b>UseCase Name</b>	11. InformDanger
<b>Description</b>	- 각층마다 엘리베이터 높이와 동일하게 위아래로 설치된 센서로부터 엘리베이터의 위치 정보를 받아와서 센서의 값이 올바르지 않을 경우 자동으로 관리자 또는 119로 SMS 구조요청이 가능하게 한다.
<b>Actors</b>	None
<b>Type</b>	Primary
<b>Cross Reference</b>	System functions R1.1.3, R3.4 UseCase : InformDanger, MoveEle
<b>Preconditions</b>	None
<b>Typical Course of Event</b>	(A) : Actor, (S) : System 1. (S) 이동시에 각층의 엘리베이터 위치를 위한 센서값을 가져옴. 2. (S) 이동시에 각층의 두 개의 센서로 엘리베이터의 위치를 파악. 3. (S) 정상적이지 못한 경우 엘리베이터의 오류로 판단. 4. (S) 관리자 혹은 119에 자동 연락.
<b>Alternative Courses of Events</b>	None
<b>Exceptional Courses of Events</b>	None

# Activity 2031. Define Essential Use Cases

## 12. DisplayTimer

<b>UseCase Name</b>	12. DisplayTimer
<b>Description</b>	- 14번 TimerCalc를 이용하여 연산되어진 데이터를 실제 입력받은 층의 현재 층수를 나타내는 디스플레이 장치옆에 사용자가 타기까지 남은 대략의 시간을 알려준다.
<b>Actors</b>	None
<b>Type</b>	Primary
<b>Cross Reference</b>	System functions R1.1.3, R2.1.1, R3.5, R3.7 Usecase : Inputbutton, DisplayTimer, TimerCalc, MoveEle
<b>Preconditions</b>	- 사용자가 외부버튼박스에 입력을 가했을 때 일어난다.
<b>Typical Course of Event</b>	(A) : Actor, (S) : System 1. (A) 사용자가 엘리베이터 외부에서 버튼입력 2. (S) 입력받은 층수의 값을 가져옴 3. (S) 엘리베이터의 현재 층수 값을 가져옴 4. (S) 중간에 미리 입력받은 층이 있으면 가져옴 5. (S) Timercalc를 이용한 평균 대기시간 계산 6. (S) 해당층에 Display 7. (S) 엘리베이터 각층 이동시마다 업데이트
<b>Alternative Courses of Events</b>	None
<b>Exceptional Courses of Events</b>	None

# Activity 2031. Define Essential Use Cases

## 13. LocationDisplay

<b>UseCase Name</b>	13. LocationDisplay
<b>Description</b>	- 현재의 엘리베이터가 어느 층에 있는지 센서를 통하여 확인하고 모든 층의 엘리베이터 입구에 이를 디스플레이 할 수 있게 한다.
<b>Actors</b>	None
<b>Type</b>	Primary
<b>Cross Reference</b>	System functions R1.1.3, R3.6 UseCase : LocationDisplay, MoveEle
<b>Preconditions</b>	None
<b>Typical Course of Event</b>	(A) : Actor, (S) : System 1. (S) 각층의 센서가 엘리베이터의 위치를 시스템에 전달. 2. (S) 해당층 표시 3. (S) 엘리베이터의 진행방향 표시 4. (S) 엘리베이터 각층 진행시마다 업데이트
<b>Alternative Courses of Events</b>	None
<b>Exceptional Courses of Events</b>	None

# Activity 2031. Define Essential Use Cases

## 14. TimerCalc

<b>UseCase Name</b>	14. TimerCalc
<b>Description</b>	- 대기시간 표현은 최근 3개월간의 사용자가 입력한 층에서부터 현재의 엘리베이터까지의 층수간의 이동소요시간을 통계 내어 상위 15%, 하위 15%의 데이터를 제외한 70%의 값을 통계 낸다.
<b>Actors</b>	None
<b>Type</b>	Primary
<b>Cross Reference</b>	System functions R2.1.1, R3.5, R3.7 UseCase : DisplayTimer, TimerCalc
<b>Preconditions</b>	- 사용자로부터 외부버튼박스의 입력을 받는다.
<b>Typical Course of Event</b>	(A) : Actor, (S) : System 1. (S) 현재엘리베이터의 위치 파악 및 사용자층과 비교 2. (S) 3개월간의 통계DB 검색 3. (S) 상·하위 15%의 결과를 제외한 평균값 산출 4. (S) 사용자에게 표시 5. (S) 타이머를 통한 대기시간 단축 표시
<b>Alternative Courses of Events</b>	None
<b>Exceptional Courses of Events</b>	Line1. 현재의 층에 엘리베이터가 있을때는 시간을 구하지 않는다.

# Activity 2031. Define Essential Use Cases

## 15. SetupFloor

<b>UseCase Name</b>	15. SetupFloor
<b>Description</b>	- 관리자가 설정할 수 있는 메뉴로 홀수층만 운행한다던지 짝수층만 운행 한다던지 하는 운행 옵션을 설정 할 수 있게 한다. - 해당층에 대해서 입력을 막는다.
<b>Actors</b>	Admin
<b>Type</b>	Secondary
<b>Cross Reference</b>	System functions R2.1.1, R4.1 UseCase : InputButton, SetupFloor
<b>Preconditions</b>	관리자 모드로 로그인이 되어있어야 한다.
<b>Typical Course of Event</b>	(A) : Actor, (S) : System 1. (A) 관리자모드로 엘리베이터 컨트롤 프로그램에 접속 2. (S) 암호입력창 출력 3. (A) 시스템암호 입력 4. (A) 층별 관리사항 변경 5. (S) 관리사항 저장 및 적용 6. (S) 해당층 버튼입력 무효화
<b>Alternative Courses of Events</b>	None
<b>Exceptional Courses of Events</b>	None

# Activity 2031. Define Essential Use Cases

## 16. SetupCamDB

<b>UseCase Name</b>	16. SetupCamDB
<b>Description</b>	- 카메라를 통한 기능과 데이터베이스에 관련된 기능들의 옵션을 설정해 주는 관리자 메뉴이다. - 카메라를 통하여 각층의 인원들을 검색할 것인지, 카메라를 통하여 위험요소를 검색할 것인지 설정 할 수 있는 기능이다. - 데이터베이스 자료를 통한 엘리베이터 자동대기 기능 등을 켜고 끌 수 있으며 어느정도의 가중치를 줄 것인지 설정할 수 있다.
<b>Actors</b>	Admin
<b>Type</b>	Secondary
<b>Cross Reference</b>	System functions R3.3, R4.2, R5.1, R5.2, R5.3, R5.4 UseCase : AutoWait, SetupCamDB, SearchGuest, SearchDanger, DatabaseCalc, DatabaseInput
<b>Preconditions</b>	관리자 모드로 로그인되어있어야 한다.
<b>Typical Course of Event</b>	(A) : Actor, (S) : System 1. (A) 관리자모드로 엘리베이터 컨트롤 프로그램에 접속. 2. (S) 암호입력창 출력 3. (A) 시스템암호 입력 4. (A) 카메라 작동여부, 인원검색여부, 위기상황 인식여부, 데이터베이스 시스템 가동여부, 자동대기 가중치 지정 등의 세부 옵션 설정 5. (S) 설정저장 6. (S) 카메라 및 자동대기 DB에 적용
<b>Alternative Courses of Events</b>	None
<b>Exceptional Courses of Events</b>	None

# Activity 2031. Define Essential Use Cases

## 17. SetupTimer

<b>UseCase Name</b>	17. SetupTimer
<b>Description</b>	- 관리자가 설정할수 있는 메뉴로 일정한 시간동안만 엘리베이터를 운행할수 있게 한다. - 정해진 시간이후에 하드웨어 전원의 자동 OFF 및 ON
<b>Actors</b>	Admin
<b>Type</b>	Secondary
<b>Cross Reference</b>	System functions R4.3 UseCase : SetupTimer
<b>Preconditions</b>	관리자 모드로 로그인이 되어있어야 한다.
<b>Typical Course of Event</b>	(A) : Actor, (S) : System 1. (A) 관리자모드로 엘리베이터 컨트롤 프로그램에 접속. 2. (S) 암호입력창 출력 3. (A) 시스템암호 입력 4. (A) 자동OFF 시간 및 자동ON 시간 설정 5. (S) 설정 저장 6. (S) ON/OFF 시간 점검 7. (S) 무게감지 및 카메라 안면인식을 통한 사용자 확인 8. (S) 엘리베이터 ON/OFF
<b>Alternative Courses of Events</b>	None
<b>Exceptional Courses of Events</b>	Line6. 설정된 시간이후 전원을 끌때에는 무게감지 및 카메라를 통한 사용자 인식을 한후에 전원을 내리도록 한다.

# Activity 2031. Define Essential Use Cases

## 18. SearchGuest

<b>UseCase Name</b>	18. SearchGuest
<b>Description</b>	- 데이터베이스에 넣을 데이터로 몇층에서 몇 명의 승객이 탔는지 Image Processing을 통하여 사람의 수를 알수 있다.
<b>Actors</b>	None
<b>Type</b>	Primary
<b>Cross Reference</b>	System functions R1.1.2, R3.3, R4.2, R5.1, R5.3 UseCase : CloseDoor, SetupCamDB, SearchGuest, DatabaseInput, AutoWait
<b>Preconditions</b>	- SetupCamDB에서 카메라를 통한 사람인식 기능을 활성화 시켜놓았다.
<b>Typical Course of Event</b>	(A) : Actor, (S) : System 1. (S) 사용자의 인식 및 카메라 기능의 활성화 여부 확인 2. (S) 문이 닫힘 3. (S) 새로운 사진 데이터 산출 4. (S) 관리자 PC로의 사진 전송 및 저장 5. (S) 사진의 잡음제거 및 분할화 6. (S) 특수 부위에 따른 얼굴 검출 및 객체생성 7. (S) 사람의 수 파악 및 다음 CloseDoor 까지 트래킹 8. (S) 새로운 사용자, 내린 사용자의 데이터 파악 9. (S) 이용객 수 및 층수, DB저장
<b>Alternative Courses of Events</b>	None
<b>Exceptional Courses of Events</b>	None

# Activity 2031. Define Essential Use Cases

## 19. SearchDanger

<b>UseCase Name</b>	19. SearchDanger
<b>Description</b>	- SearchGuest를 통하여 찾은 대상이 일정시간 이상 엘리베이터에 있거나 급격한 이동을 할 경우에 이를 위험 하다고 판단하여 관리자 또는 관리소등에 연락 할 수 있게 한다.
<b>Actors</b>	None
<b>Type</b>	Primary
<b>Cross Reference</b>	System functions R2.2, R4.2, R5.2 UseCase : CallAdmin, SetupCamDb, SearchDanger
<b>Preconditions</b>	- SetupCamDB에서 카메라를 통한 위험인식 기능을 활성화 해 놓았다.
<b>Typical Course of Event</b>	(A) : Actor, (S) : System 1. (S) 위험 인식 및 카메라 기능의 활성화 여부 확인 2. (S) UseCase 18의 찾은 사용자 포인터의 내부에서의 오랜 대기 및 엄청난 이동의 감지 3. (S) 관리자 호출
<b>Alternative Courses of Events</b>	None
<b>Exceptional Courses of Events</b>	None

# Activity 2031. Define Essential Use Cases

## 20. DatabaseInput

<b>UseCase Name</b>	20. DatabaseInput
<b>Description</b>	- SearchGuest를 통하여 얻은 사람들이 탑승한 층과 사람수, 인원이동에 따른 경과시간, 작동시간 등을 데이터베이스에 저장한다.
<b>Actors</b>	None
<b>Type</b>	Primary
<b>Cross Reference</b>	System functions R1.1.2, R5.1, R4.2, R5.4 UseCase : CloseDoor, SearchGuest, DatabaseCalc, SetupCamDb
<b>Preconditions</b>	- 사용자의 데이터, 이동층수 및 시간 데이터가 산출되어져 있다.
<b>Typical Course of Event</b>	(A) : Actor, (S) : System 1. (S) DB에 저장되기 위한 데이터가 산출되어 있다. 2. (S) 날짜, 사람 수, 시간, 이미지, 등을 설정된 형태로 DB에 저장한다. 3. (S) 데이터통계를 위한 자료로 사용된다.
<b>Alternative Courses of Events</b>	None
<b>Exceptional Courses of Events</b>	None

# Activity 2031. Define Essential Use Cases

## 21. DatabaseCalc

<b>UseCase Name</b>	21. DatabaseCalc
<b>Description</b>	- 관리자가 설정한 시간동안의 데이터베이스 자료를 엘리베이터 컨트롤러가 사용할 수 있는 데이터로 변환 할 수 있게 한다. - 관리자가 직접 조작 할 수 있고 평소에는 컨트롤러 내부적으로 데이터베이스의 값을 가져다 쓸 수 있다. - 대기시간 표현이나 자동대기 기능을 위하여 쓸 데이터를 가공하는 기능
<b>Actors</b>	None
<b>Type</b>	Secondary
<b>Cross Reference</b>	System functions R3.3, R4.2, R5.3, R5.4 UseCase : AutoWaitm SetupCamDb, DatabaseCalc, DatabaseInput
<b>Preconditions</b>	None
<b>Typical Course of Event</b>	(A) : Actor, (S) : System 1. (A) 통계별 기간설정 2. (A) 통계방법 및 옵션 설정 3. (S) 모든 데이터의 평균 산출 4. (S) 산출된 데이터 적용 및 저장
<b>Alternative Courses of Events</b>	None
<b>Exceptional Courses of Events</b>	None



# Activity 2033. Define Conceptual Model

## ❖ Using Concept Category List

Concept Category	Domain Class
Physical of tangible objects	센서, 카메라, 버튼박스
Specifications, designs, of descriptions of things	시간, 적재무게, DB-Data, 운행정보, 대기층
Places	현재층
Transactions	X
Roles of people	사용자, 관리자
Containers of other things	엘리베이터
Things in a container	운행정보, 사용자
organizations	X

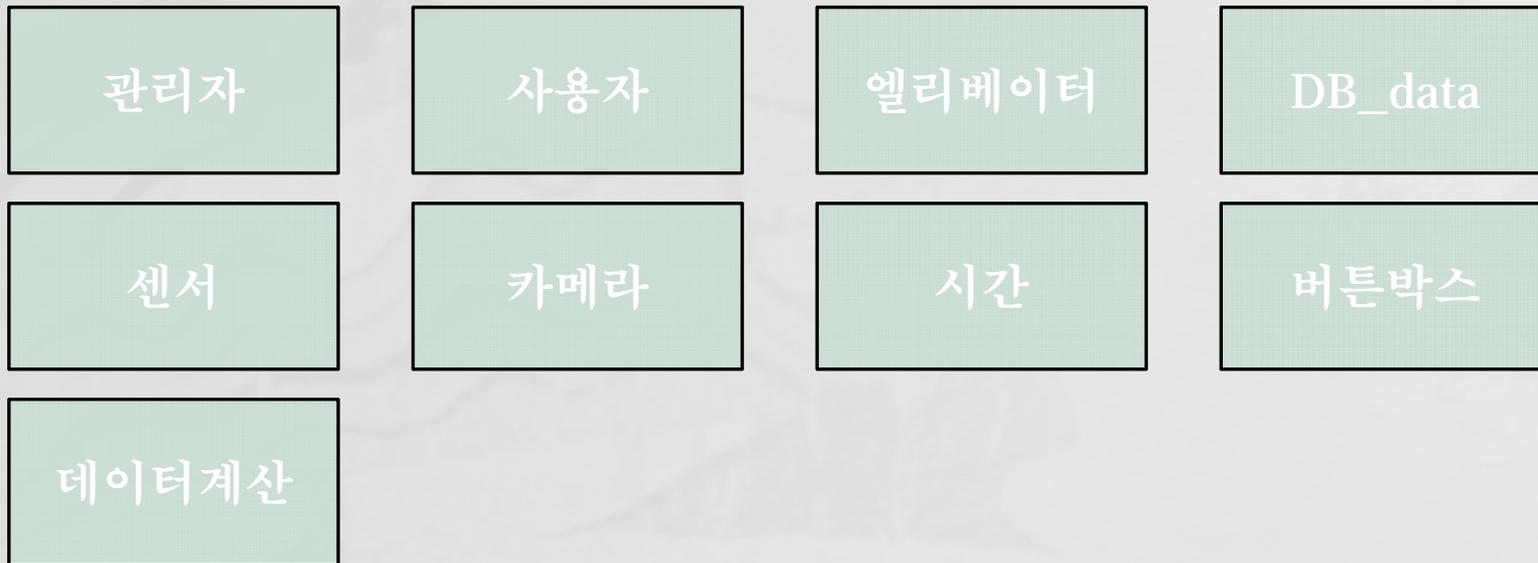
# Activity 2033. Define Conceptual Model

## ❖ Using Noun Phrases



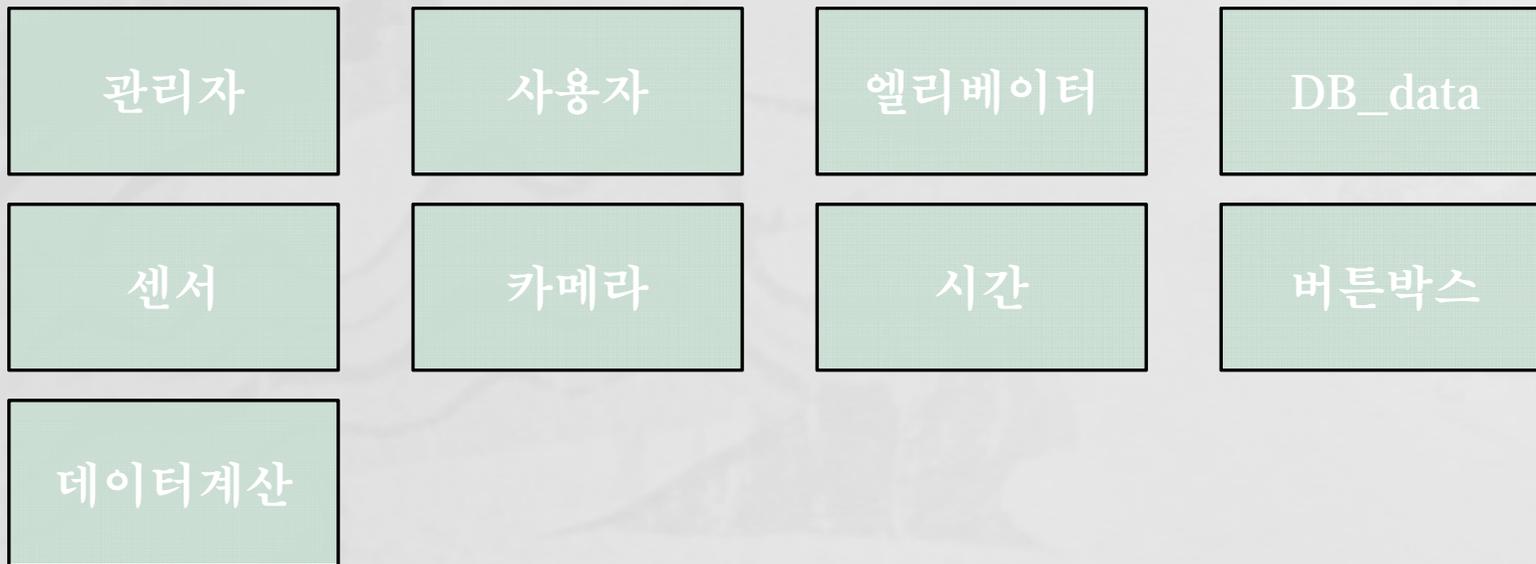
# Activity 2033. Define Conceptual Model

❖ Step2. Assign class name into a concept



# Activity 2033. Define Conceptual Model

❖ Step3. Draw a conceptual class diagram



# Activity 2033. Define Conceptual Model

## ❖ Step4. Identify and add associations Use 'Association Category List

Association Category	Associations
A is known/logged/recorded/reported/captured in B	Administrator – Sensor Administrator – Elevator Administrator – Camera Elevator – DB_calc DB_calc – DB_data
A is a line item of B	Elevator – Timer DB_data – Timer DB_calc - Timer
A is recorded in B	Elevator – DB_calc DB_data – Passenger DB_data - Camera
A is Related to a transaction of B	Administrator – Sensor Administrator – Camera Administrator – DB_data
A is an organization submit of B	Elevator - ButtonBox

# Activity 2033. Define Conceptual Model

## ❖ Step5. Assign priority into associations

Association Name	Priority
Elevator - Timer	high
DB_data - Timer	high
DB_calc - Timer	high
Elevator - DB_calc	high
DB_data - Passenger	high
DB_data - Camera	high
DB_calc – DB_data	high
Administrator - Sensor	high
Administrator - Camera	high
Administrator - DB_data	high
Elevator - ButtonBox	high

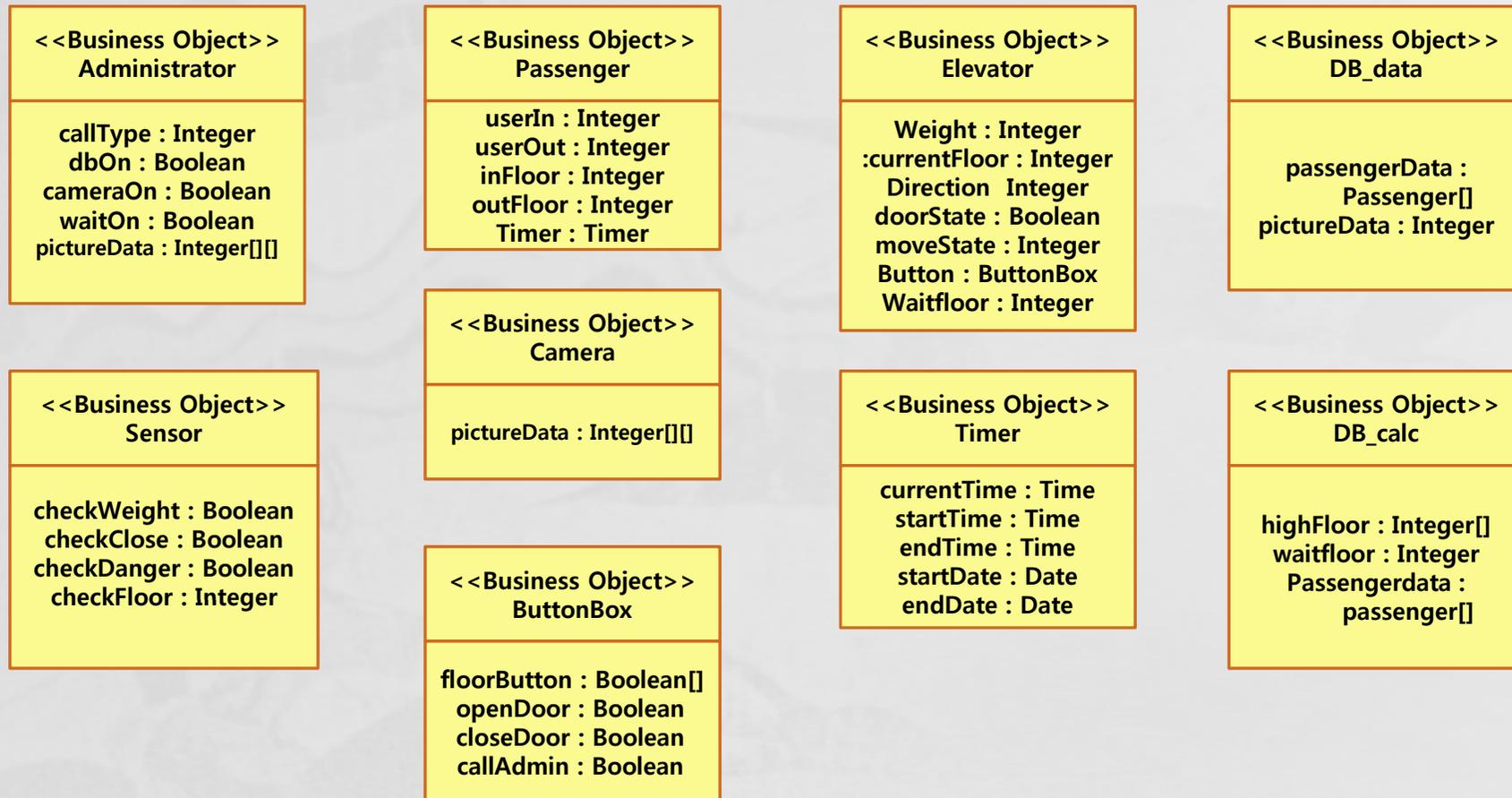
# Activity 2033. Define Conceptual Model

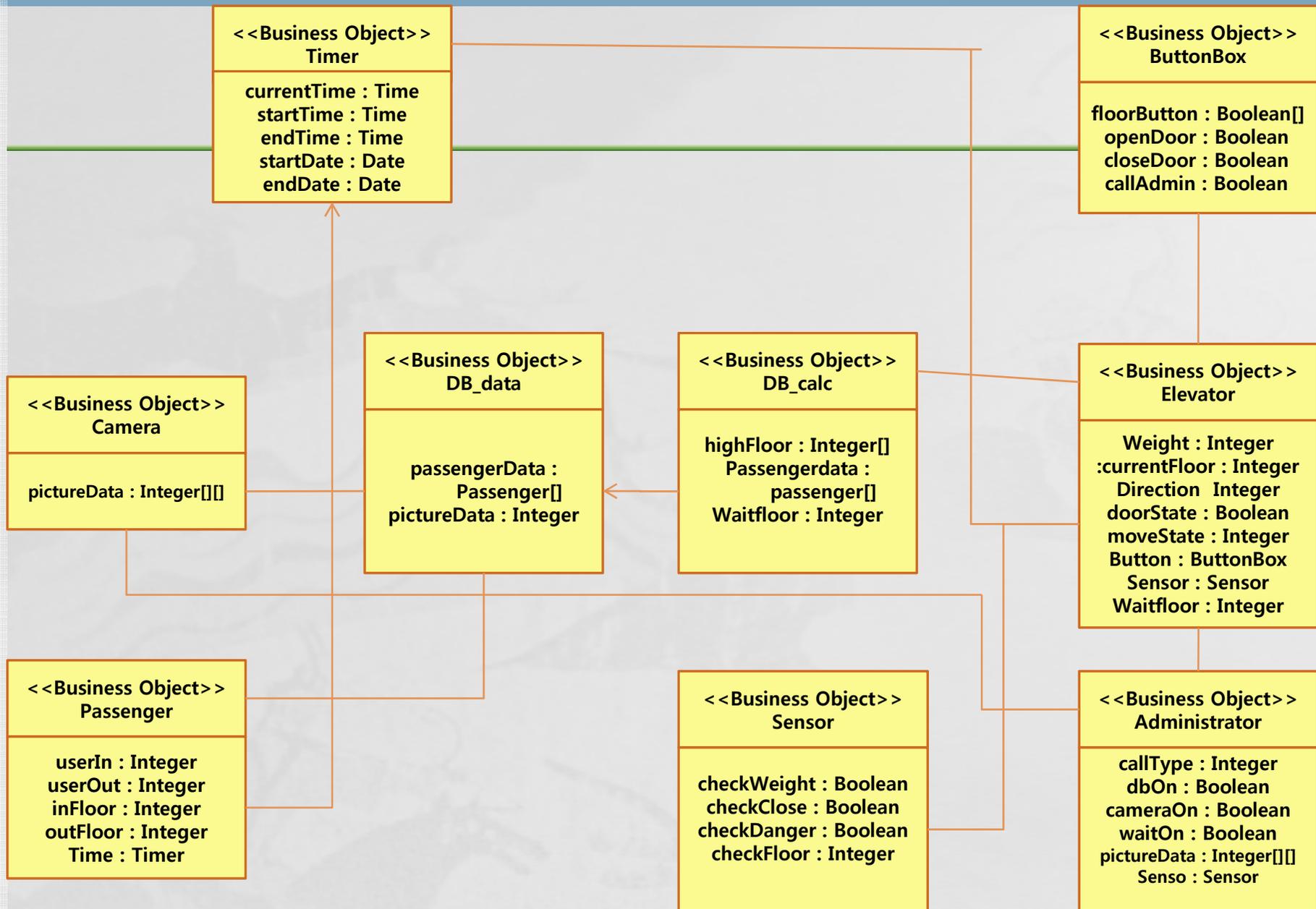
## ❖ Step6. Assign name into associations

- Administrator는 Camera를 참조한다.
- Administrator는 Elevator를 참조한다.
- Elevator 는 ButtonBox를 가지고 있다.
- Elevator 는 Timer를 참조한다.
- Elevator 는 Sensor를 참조한다.
- Passenger는 Timer를 가지고 있다.
- DB\_DATA는 Camera를 복사한다.
- DB\_DATA는 Passenger를 복사한다.
- DB\_Calc는 DB\_data를 참조한다.

# Activity 2033. Define Conceptual Model

## ❖ Using Concept Category List





# Activity 2034. Refine Glossary

## ❖ Refine Terms

Term	Category	remarks
Administrator	class	다른클래스들을 키고 끄는 기능을 한다. 호출을 받음
Passenger	class	타고내리는 층/인원수/시간을 구조체로 받아서 DB_calc로전달
Elevator	class	엘리베이터 운행에 관련된 클래스
ButtonBox	class	버튼 입력에 관련된 클래스
DB_data	class	수집된 데이터를 저장하는 역할을 하는 클래스
DB_calc	class	DB_data로부터 정보를 받아서 가공하는 클래스
Sensor	class	각센서의 이상 유무를 확인하는 클래스
Camera	class	카메라 작동에 관련된 클래스
Timer	class	현재/시작/도착시간, 날짜등 시간에 관련된 정보를 얻는 클래스
Administrator.callType	Attribute	관리자 호출의 이유를 설정하는 변수. 숫자에 따라 설정할 수 있다
Administrator.dbOn	Attribute	db통계클래스를 활성화/비활성화 시키는 변수
Administrator.cameraOn	Attribute	camera클래스를 활성화/비활성화 시키는 변수
Administrator.waitOn	Attribute	자동대기기능을 활성화/비활성화 시키는 변수
Administrator.pictureData	Attribute	관리자가 캡쳐 화면을 볼수있도록 데이터를 받아오는 변수
Administrator.sensorOn	Attribute	Elevator가 가진 센서값을 받아오는 변수
Passenger.userIn	Attribute	해당층의 탑승한 고객의 수를 저장하는 변수
Passenger.userOut	Attribute	해당층의 하차한 고객의 수를 저장하는 변수
Passenger.inFloor	Attribute	고객이 탑승한 층을 저장하는 변수
Passenger.outFloor	Attribute	고객이 하차한 층을 저장하는 변수

# Activity 2034. Refine Glossary

## ❖ Refine Terms

Term	Category	remarks
Elevator.weight	Attribute	무게허용한도를 가지고 있는 변수
Elevator.currentFloor	Attribute	현재층의 데이터를 가지고 있는 변수
Elevator.direction	Attribute	엘리베이터의 이동방향을 가지고 있는 변수
Elevator.doorState	Attribute	문의 상태값을 가지고 있는 변수
Elevator.moveState	Attribute	이동상태를 가지고 있는 변수
Elevator.button	Attribute	입력된 버튼의 정보를 가지고있는 구조체형 변수
Elevator.sensor	Attribute	sensor클래스의 정보를 받아오는 변수
Elevator.waitFloor	Attribute	자동대기기능을 실행할 층의 정보를 가지고 있는 변수
ButtonBox.floorButton	Attribute	버튼박스의 층버튼 입력 정보를 가진 배열형 변수
ButtonBox.openDoor	Attribute	버튼박스의 문열기 기능 버튼 입력 정보를 가진 변수
ButtonBox.closeDoor	Attribute	버튼박스의 문닫기 기능 버튼 입력 정보를 가진 변수
ButtonBox.callAdmin	Attribute	버튼박스의 관리자 호출 기능 버튼 입력 정보를 가진 변수
DB_data.passengerData	Attribute	승객 이용정보를 가지고있는 구조체 변수
DB_data.pictureData	Attribute	사진정보를 가지고 있는 변수
DB_calc.highFloor	Attribute	자동대기층 우선순위를 계산해서 가지고있는 배열형 변수
DB_calc.passengerData	Attribute	DB_data에서 통계 계산을 위해 가져오는 구조체 변수
DB_calc.waitFloor	Attribute	현재 자동대기층값을 가지고 있는 변수
Sensor.checkWeight	Attribute	중량초과의 유무를 가지고 있는 변수
Sensor.checkClose	Attribute	문닫힘 상태 유무를 가지고 있는 변수
Sensor.checkDanger	Attribute	위험상태 유무를 가지고 있는 변수
Sensor.checkFloor	Attribute	엘리베이터의 위치를 체크하여 가지고 있는 변수

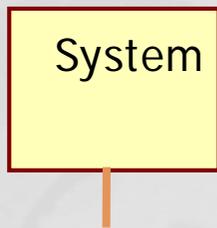
# Activity 2034. Refine Glossary

## ❖ Refine Terms

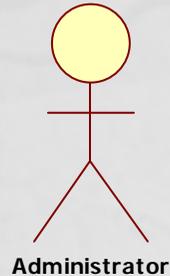
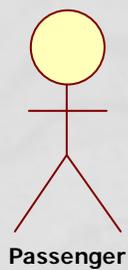
Term	Category	remarks
Camera.pitureData	Attribute	현재 카메라로 찍은 데이터값을 가진 변수
Timer.currentTime	Attribute	현재 시간값을 가진 변수
Timer.startTime	Attribute	운영 시작시간값을 가진 변수
Timer.endTime	Attribute	운영 종료시간값을 가진 변수
Timer.startDate	Attribute	통계기간 설정에 필요한 시작 날짜를 가진 변수
Timer.endDate	Attribute	통계기간 설정에 필요한 종료 날짜를 가진 변수

# Activity 2035. Define System Sequence Diagrams

## ❖ Step 1

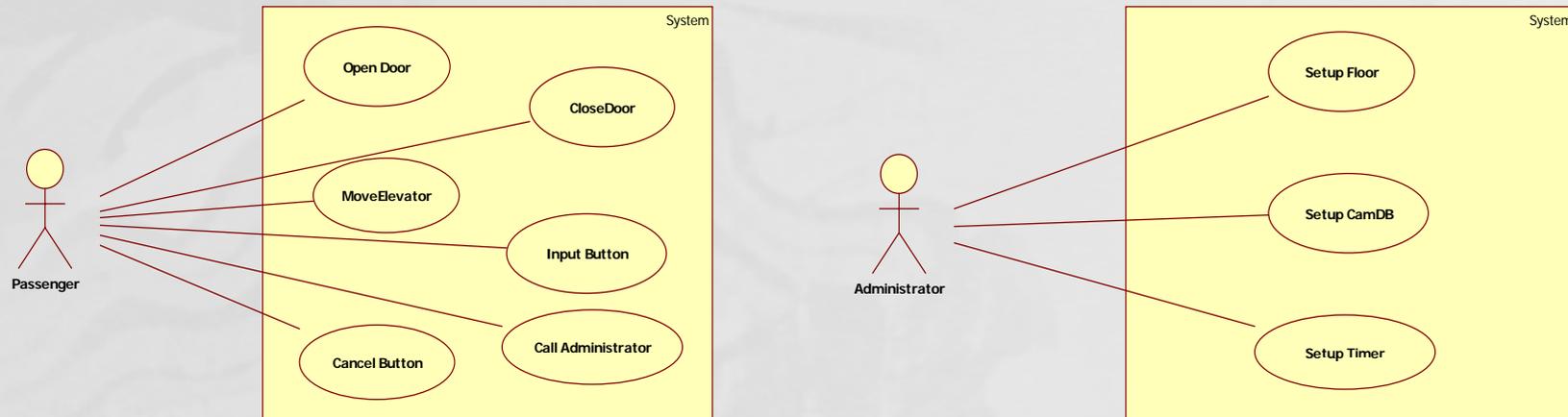


## ❖ Step 2



# Activity 2035. Define System Sequence Diagrams

## ❖ Step 3. Determine System Boundary

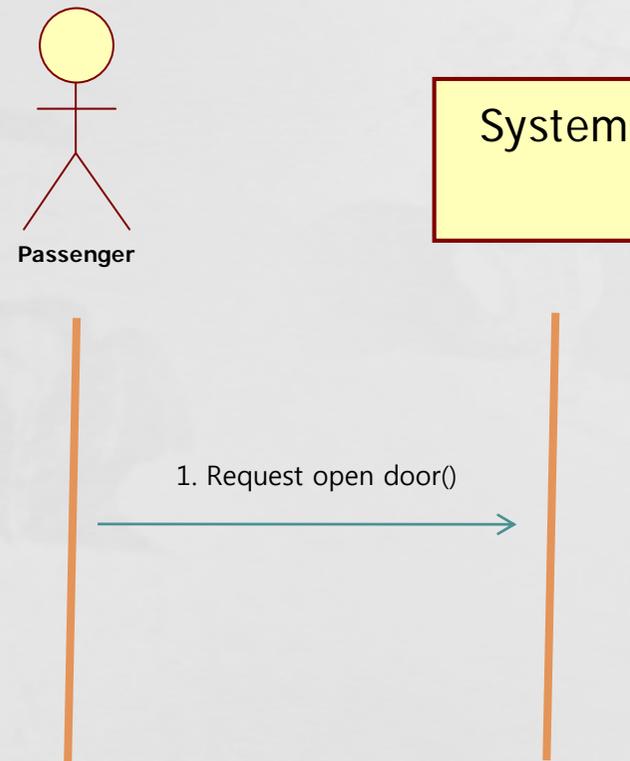


# Activity 2035. Define System Sequence Diagrams

## ❖ Step 3. Determine System Boundary

### Use Case: 1. Open Door

1. 사용자는 외부 버튼 박스의 화살표 키를 누르거나 엘리베이터 내부의 버튼 박스의 층 버튼 또는 문열림 버튼을 입력.
2. 엘리베이터가 운행 중이지 않을 경우 MoveEle를 통하여 내·외부의 입력 받은 층으로 이동.
3. 입력 받은 층에 도착했을 시에 문을 옴.
4. 처리한 층에 대한 입력버튼 혹은 화살표 버튼의 입력을 없앴.
5. 일정 시간 문이 열린 시간이 지나면 문을 닫는다.

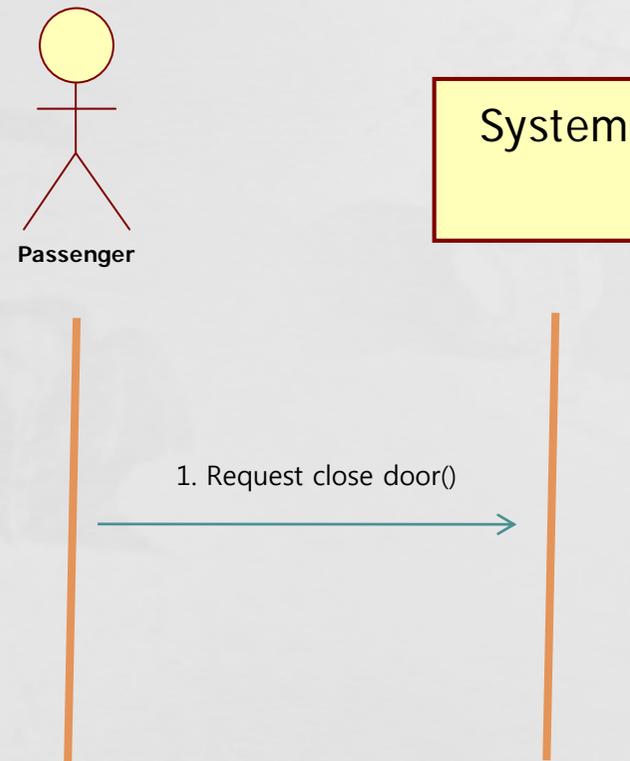


# Activity 2035. Define System Sequence Diagrams

## ❖ Step 3. Determine System Boundary

### Use Case: 2. Close Door

- 1.1. 문닫힘 버튼을 선택한다.
- 1.2. 문이 열려져있는 상태가 지속됨을 인지한 UseCase1로부터 명령을 받는다.
2. 문을 닫기전 엘리베이터에서 수용할 수 있는 무게가 엘리베이터에 탔는지 점검후 허용되지 않으면 문을 닫지 않는다.
3. 문이 완전히 닫히기 전까지 문사이의 사물파악 센서를 통하여 중간에 끼거나 또다른 사용자가 접근 중인지에 대하여 파악.
4. 해당 층수 버튼의 입력으로부터 엘리베이터의 이동시간이 얼마만큼 오래 걸렸는지 파악.
5. 카메라를 통하여 이전에 타고 있던 사용자 포인터가 사라진 수를 체크.
6. 카메라를 통하여 새로 탑승한 사용자의 수를 파악.
7. 4~6번의 데이터를 DB에 입력.

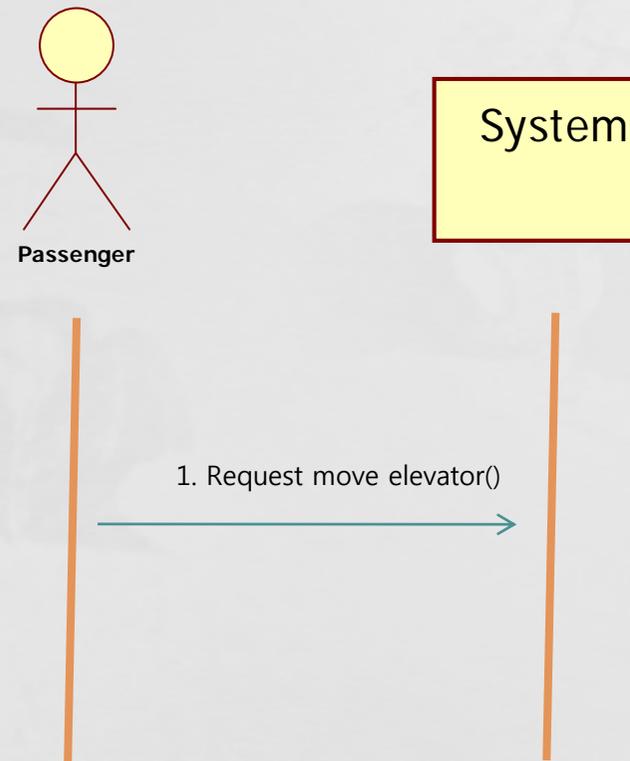


# Activity 2035. Define System Sequence Diagrams

## ❖ Step 3. Determine System Boundary

### Use Case: 3. MoveEle

- 1.1. 사용자가 목적지 층버튼 입력.
- 1.2. 사용자들의 입력이 전무하고 엘리베이터 내부에 사람이 없을 때 자동 대기할 층의 정보를 입력.
2. 문이 닫혀 있는지 점검한 후 엘리베이터를 이동.
3. 이동 중에 외부대기자의 대기시간을 업데이트
4. 위치센서를 통한 현재의 위치 업데이트
5. 이동 중에 InformDanger를 실행

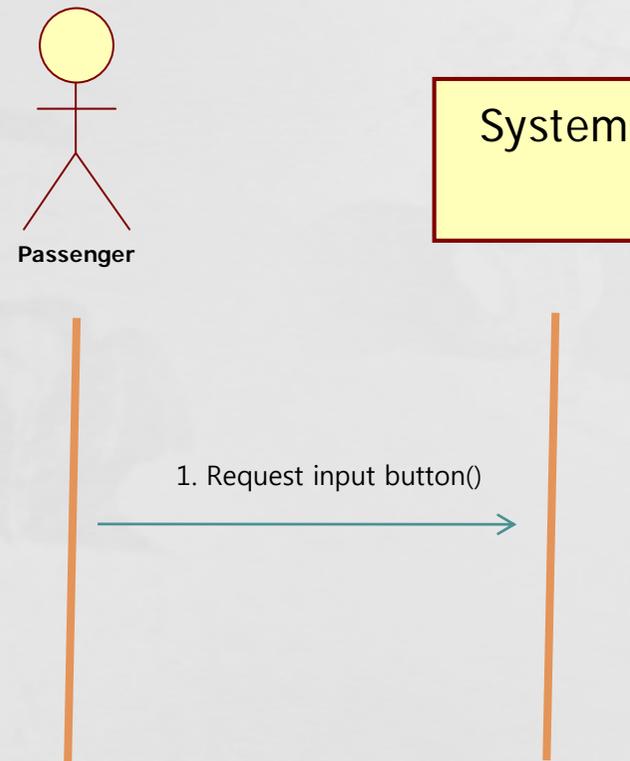


# Activity 2035. Define System Sequence Diagrams

## ❖ Step 3. Determine System Boundary

### Use Case: 4. Input Button

- \* 버튼을 입력.  
-> 버튼박스가 여러개일 경우 멀티인풋 기능을 실행.
- \* 원하는 층수버튼 또는 화살표 버튼을 입력.  
-> 해당 층으로 이동.
- \* 문열림 또는 문닫힘 버튼을 입력.  
-> 문열림 또는 문닫힘 기능을 실행.
- \* 관리자 호출 버튼을 입력.  
-> 관리자호출 기능을 실행.
- \* 외부 화살표 버튼 입력  
-> 대기예상시간 표시

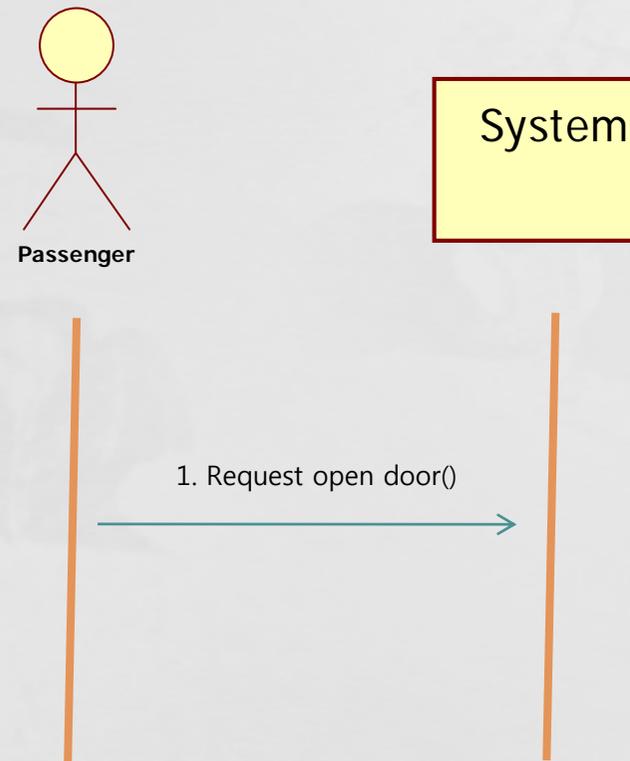


# Activity 2035. Define System Sequence Diagrams

## ❖ Step 3. Determine System Boundary

### Use Case: 1. Open Door

1. 사용자는 외부 버튼 박스의 화살표 키를 누르거나 엘리베이터 내부의 버튼 박스의 층 버튼 또는 문열림 버튼을 입력.
2. 엘리베이터가 운행 중이지 않을 경우 MoveEle를 통하여 내·외부의 입력 받은 층으로 이동.
3. 입력 받은 층에 도착했을 시에 문을 옴.
4. 처리한 층에 대한 입력버튼 혹은 화살표 버튼의 입력을 없앴.
5. 일정 시간 문이 열린 시간이 지나면 문을 닫는다.

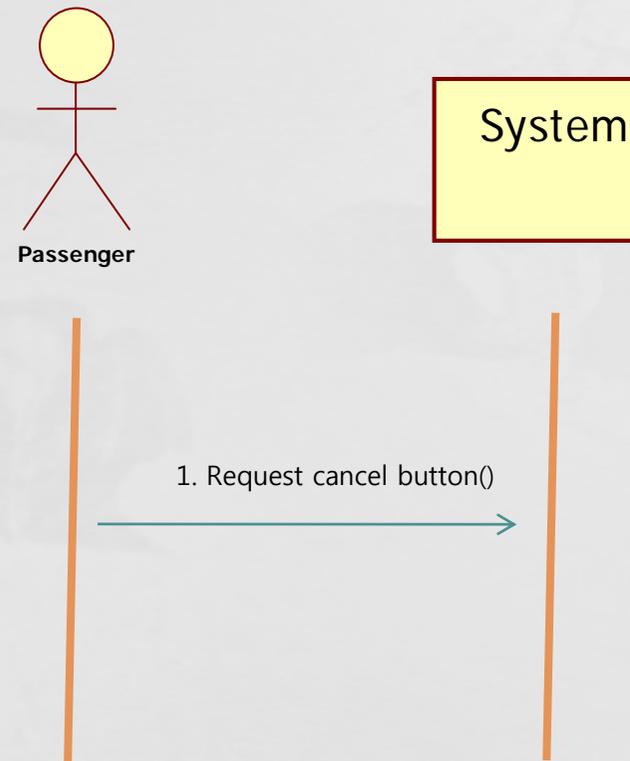


# Activity 2035. Define System Sequence Diagrams

## ❖ Step 3. Determine System Boundary

### Use Case: 5. Cancel Button

1. 현재 입력된 층수를 취소.
2. 버튼박스의 입력내용을 취소.
3. 모든 버튼박스의 내용을 수정.
4. 현재 엘리베이터의 진행방향을 판단하여 운행, 정지, 역방향 운행 등을 실행.

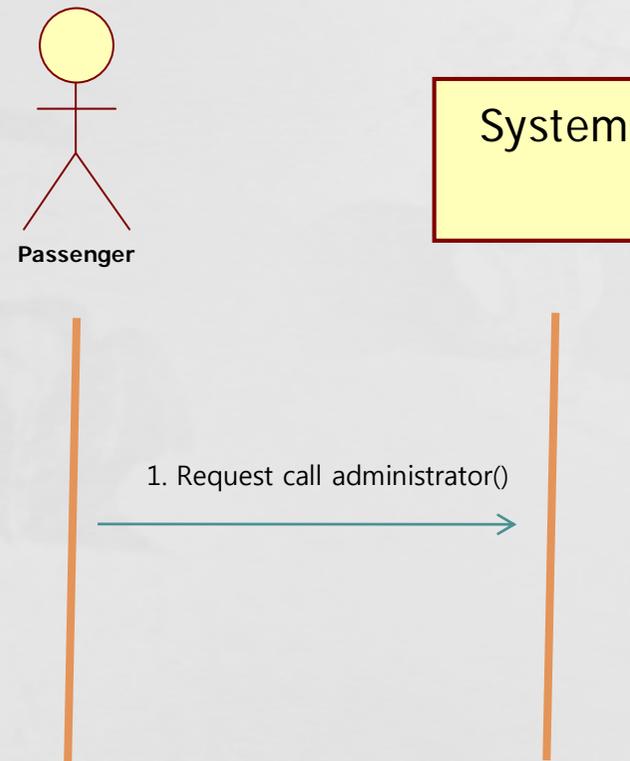


# Activity 2035. Define System Sequence Diagrams

## ❖ Step 3. Determine System Boundary

### Use Case: 6. Call Admin

1. 사용자가 관리자 버튼을 입력.
2. 인풋박스에서 관리자 호출 기능을 실행.
3. 관리자가 설정한 방법으로 호출을 실시.

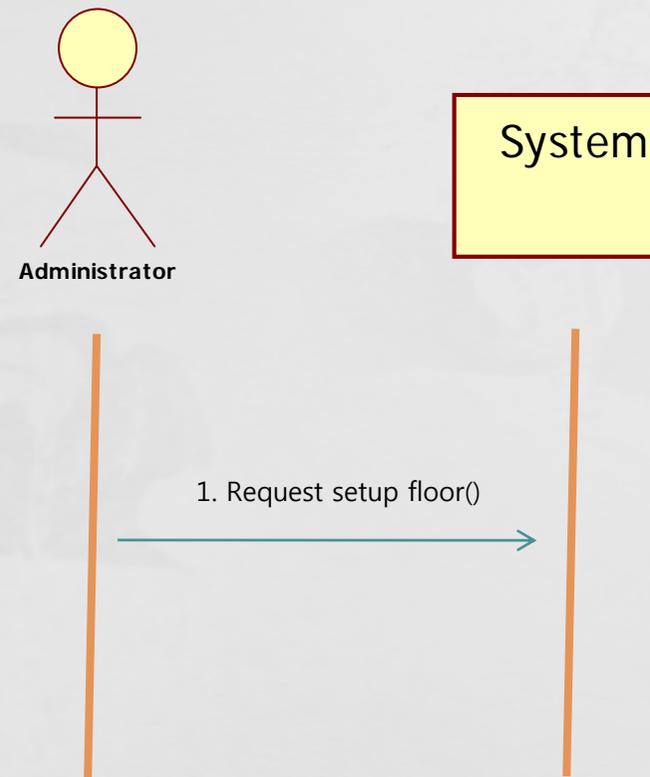


# Activity 2035. Define System Sequence Diagrams

## ❖ Step 3. Determine System Boundary

### Use Case: 7. Setup Floor

1. 관리자모드로 엘리베이터 컨트롤 프로그램에 접속
2. 암호 입력창 출력
3. 시스템암호 입력
4. 층별 관리사항 변경
5. 관리사항 저장 및 적용
6. 해당 층 버튼입력 무효화

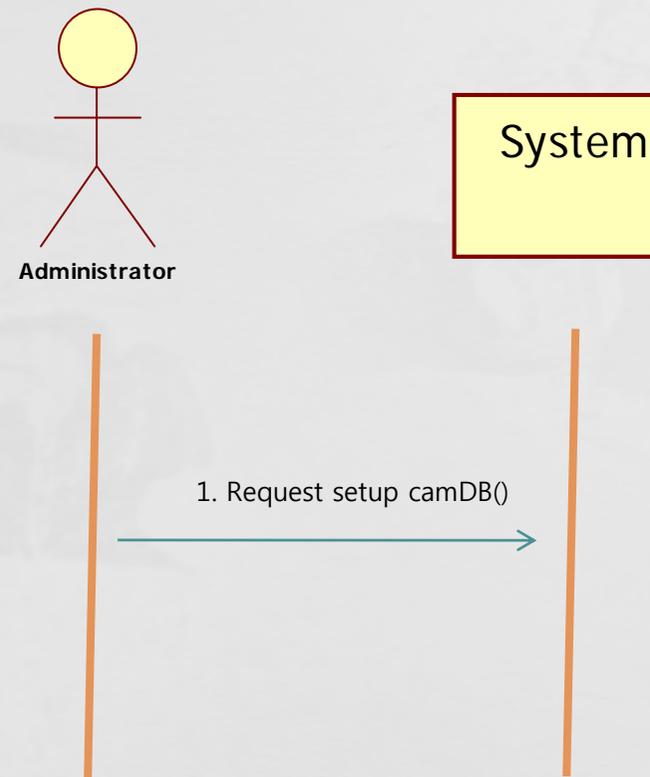


# Activity 2035. Define System Sequence Diagrams

## ❖ Step 3. Determine System Boundary

### Use Case: 8. Setup CamDB

1. 관리자모드로 엘리베이터 컨트롤 프로그램에 접속.
2. 암호 입력창 출력
3. 시스템암호 입력
4. 카메라 작동여부, 인원검색여부, 위기상황 인식여부, 데이터베이스 시스템 가동여부, 자동대기 가중치 지정 등의 세부 옵션 설정
5. 설정저장
6. 카메라 및 자동대기 DB에 적용

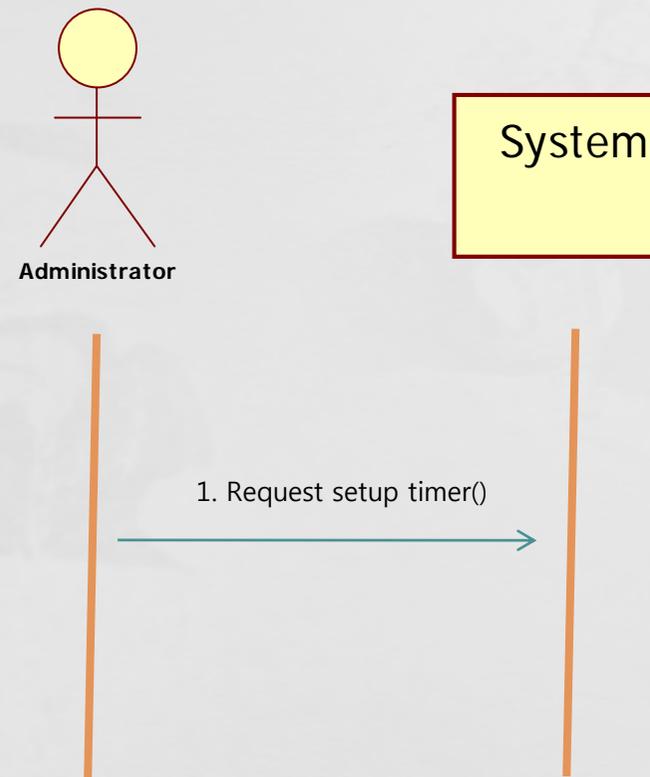


# Activity 2035. Define System Sequence Diagrams

## ❖ Step 3. Determine System Boundary

### Use Case: 9. Setup Timer

1. 관리자모드로 엘리베이터 컨트롤 프로그램에 접속.
2. 암호입력창 출력
3. 시스템암호 입력
4. 자동OFF 시간 및 자동ON 시간 설정
5. 설정 저장
6. ON/OFF 시간 점검
7. 무게감지 및 카메라 안면인식을 통한 사용자 확인
8. 엘리베이터 ON/OFF



# Activity 2036. Define Operation Contracts

## Step1. Identify system operation from system sequence diagrams

Use Case	Name of Actor-Activated Event	System Operations
1.Opendoor	1: Request open door()	1. openDoor()
2.Closeddoor	1: Request open door()	2. closeDoor()
3.Moveele	1: Request move elevator()	3. moveEle()
4.InputButton	1: Request input button()	4. inputButton()
5.CancelButton	1: Request cancel button()	5. cancelButton()
6.CallAdmin	1: Request call administrator()	6. callAdmin()
7.MultiInput	N/A	N/A
8.WeightConfirm	N/A	N/A
9.SafeDoor	N/A	N/A
10.AutoWait	N/A	N/A
11.InformDanger	N/A	N/A
12.DisplayTimer	N/A	N/A
13.LocationDisplay	N/A	N/A
14.TimerCalc	N/A	N/A
15.SetupFloor	1: Request setup floor()	7. setupFloor()
16.SetupCamDB	1: Request setup CamDB()	8. setupCamDB()
17.SetupTimer	1: Request setup Timer	9. setupTimer()
18.SearchGuest	N/A	N/A
19.SearchDanger	N/A	N/A
20.DatabaseInput	N/A	N/A
21.DatabaseCalc	N/A	N/A

# Activity 2036. Define Operation Contracts

Step2. Fill contracts according to the format below

<b>Name</b>	openDoor()
<b>Responsibilities</b>	N/A
<b>Type</b>	System
<b>Cross References</b>	System functions R1.1.1, R1.1.2, R1.1.3, R2.1.1 UseCase : CloseDoor, MoveEle, InputButton
<b>Notes</b>	
<b>Exceptions</b>	사용자가 문열림 버튼 또는 외부의 화살표 버튼을 입력했을시에는 문이 열린시간을 재는 타이머를 초기화 시킨다.
<b>Output</b>	엘리베이터의 문을 연다
<b>Pre-conditions</b>	N/A
<b>Post-conditions</b>	층버튼/혹은 문열림 입력이 해제된다.

# Activity 2036. Define Operation Contracts

## Step2. Fill contracts according to the format below

<b>Name</b>	closeDoor()
<b>Responsibilities</b>	요청이 없어도 지정된 시간이 지나면 자동으로 닫혀야 한다. 문닫힘 기능이 수행되는동안 물체감지 센서를 작동시켜야 한다.
<b>Type</b>	System
<b>Cross References</b>	System functions R1.1.1, R1.1.2, R2.1.1, R3.1, R3.2, R5.1, R5.3 UseCase : OpenDoor, CloseDoor, InputButton, WeightConfirm SafeDoor, SearchGuest, DatabaseInput
<b>Notes</b>	
<b>Exceptions</b>	문 닫힘 버튼과 문 열림 버튼의 동시 입력 시에는 문 열림 버튼에 우선권을 부여한다.
<b>Output</b>	엘리베이터의 문을 닫는다
<b>Pre-conditions</b>	문이 열린 상태에서만 동작한다
<b>Post-conditions</b>	엘리베이터의 이동층/시간을 파악하여 DB로 보낸다. 카메라를 통하여 하차인원을 파악하여 DB로 보낸다. 카메라를 통하여 승차인원을 파악하여 DB로 보낸다. 자동대기 기능을 실행할지를 감지하고 실행한다.

# Activity 2036. Define Operation Contracts

Step2. Fill contracts according to the format below

<b>Name</b>	moveEle()
<b>Responsibilities</b>	이동중에는 InformDanger를 실행해야 한다. 이용자가 없을때는 자동대기기능 실행을 통해 이동해야 한다.
<b>Type</b>	System
<b>Cross References</b>	System functions R1.1.3, R2.1.1, R3.3, R3.4, R3.5, R3.6 UseCase : OpenDoor, InputButton, DisplayTimer, LocationDisplay, InformDanger, AutoWait
<b>Notes</b>	
<b>Exceptions</b>	이동시에 문열림 혹은 문닫힘 버튼의 입력은 무효화한다.
<b>Output</b>	해당층으로 이동한다.
<b>Pre-conditions</b>	문닫힘 상태에 있을때에만 가능하다.
<b>Post-conditions</b>	문열림 기능을 수행한다.

# Activity 2036. Define Operation Contracts

## Step2. Fill contracts according to the format below

<b>Name</b>	inputButton()
<b>Responsibilities</b>	관리자 호출버튼을 눌렀을 시에는 버튼박스에 표시하지 않아야한다. 버튼박스의 개수가 n개일 경우 하나의 입력에 대해서 multiInput기능을 수행하여 모든 버튼박스에 표시하여야 한다. 외부버튼 박스를 눌렀을 경우 databasercalc기능을 적용하여 Display에 예상 대기시간을 표시해 주어야 한다.
<b>Type</b>	System
<b>Cross References</b>	System functions R1.1.1, R1.1.2, R1.1.3, R2.1.1, R2.1.2, R2.2, R2.3 ,R3.5 UseCase : OpenDoor, CloseDoor, MoveEle, InputButton, CancelButton, Call Admin, MultiInput, DisplayTimer
<b>Notes</b>	
<b>Exceptions</b>	현재 엘리베이터가 위치한 층을 입력 했을 시에는 버튼 입력을 취소한다.
<b>Output</b>	해당 입력에 대한 내용을 실행한다.
<b>Pre-conditions</b>	None
<b>Post-conditions</b>	입력 내용을 하드웨어 장비에게 보낸다. 버튼박스에 해당입력내용을 표시한다.

# Activity 2036. Define Operation Contracts

## Step2. Fill contracts according to the format below

<b>Name</b>	cancelButton()
<b>Responsibilities</b>	버튼박스의 개수가 n개일 경우 하나의 입력에 대해서 multiInput기능을 수행하여 모든 버튼박스에 표시된 내용을 없애야 한다.
<b>Type</b>	System
<b>Cross References</b>	System functions R1.1.3, R2.1.1, R2.1.2, R2.3 UseCase : MoveEle, MultiInput, InputButton
<b>Notes</b>	
<b>Exceptions</b>	관리자 호출, 문닫힘, 문열림의 경우 취소가 없다.
<b>Output</b>	입력되어있던 inputButton()의 실행내용을 취소한다.
<b>Pre-conditions</b>	버튼이 현재 입력되어있다.
<b>Post-conditions</b>	버튼박스에 표시되어있던 입력내용을 없앤다.

# Activity 2036. Define Operation Contracts

Step2. Fill contracts according to the format below

<b>Name</b>	callAdmin()
<b>Responsibilities</b>	searchDanger기능을 통해서도 자동으로 실행 가능하여야 한다.
<b>Type</b>	System
<b>Cross References</b>	System functions R2.1.1, R2.2 UseCase : InputButton, CallAdmin
<b>Notes</b>	
<b>Exceptions</b>	None
<b>Output</b>	시스템에 설정된 방법으로 관리자에게 연락한다. (인터폰, sms 등)
<b>Pre-conditions</b>	None
<b>Post-conditions</b>	searchDanger기능을 통해서 실행된 경우 moveEle의 기능을 막는다.

# Activity 2036. Define Operation Contracts

Step2. Fill contracts according to the format below

<b>Name</b>	setupFloor()
<b>Responsibilities</b>	비밀번호가 DB의 정보와 일치해야 조작성 가능하다.
<b>Type</b>	System
<b>Cross References</b>	System functions R2.1.1, R4.1 UseCase : InputButton, SetupFloor
<b>Notes</b>	
<b>Exceptions</b>	None
<b>Output</b>	층별 관리사항이 변경된다.
<b>Pre-conditions</b>	관리자로 로그인 되어있어야 한다.
<b>Post-conditions</b>	해당층 버튼 입력 무효화

# Activity 2036. Define Operation Contracts

Step2. Fill contracts according to the format below

<b>Name</b>	setupCamDB()
<b>Responsibilities</b>	비밀번호가 DB의 정보와 일치해야 조작성 가능하다.
<b>Type</b>	
<b>Cross References</b>	System functions R3.3, R4.2, R5.1, R5.2, R5.3, R5.4 UseCase : AutoWait, SetupCamDB, SearchGuest, SearchDanger, DatabaseCalc, DatabaseInput
<b>Notes</b>	
<b>Exceptions</b>	None
<b>Output</b>	엘리베이터 운행의 세부옵션이 변경되고 카메라/자동대기 DB에 적용된다.
<b>Pre-conditions</b>	관리자 모드로 로그인 되어있어야 한다.
<b>Post-conditions</b>	카메라 작동여부 변경 인원 검색여부 변경 위기상황 인식여부 변경 DB System가동여부 변경 자동대기 가중치 지정/실행 여부 변경

# Activity 2036. Define Operation Contracts

Step2. Fill contracts according to the format below

<b>Name</b>	setupTimer()
<b>Responsibilities</b>	비밀번호가 DB의 정보와 일치해야 조작성 가능하다.
<b>Type</b>	System
<b>Cross References</b>	System functions R4.3 UseCase : SetupTimer
<b>Notes</b>	
<b>Exceptions</b>	설정된 시간 이후 전원을 끌 때에는 무게감지 및 카메라를 통한 사용자 인식을 한 후에 전원을 내리도록 한다.
<b>Output</b>	설정시간 이후의 하드웨어 전원 자동 On/Off
<b>Pre-conditions</b>	관리자 모드로 로그인 되어있어야 한다.
<b>Post-conditions</b>	현재 설정을 저장한다.

# Activity 2037. Define Stats Diagrams

< State Diagram for “Button Box” >    < State Diagram for “Elevator” >

