

# Failover Elevator Controller Simulator

## Phase 2130. Analyze

### Activity 2131. Define Essential Use Cases

Use Case	1. 시뮬레이션 시작
Actor	User
Purpose	전체 시뮬레이션이 작동중이지 않으면 작동시킨다.
Overview	전체 시뮬레이션이 작동 중이지 않으면 작동 시키고 승객을 생성하기 시작한다.
Type	Secondary
Cross Reference	Functions: R1.1, R1.2, R3.5
Pre-Requisites	시뮬레이션이 작동하고 있지 않아야함.
Typical Courses of Events	(A) : Actor, (S) : System <ol style="list-style-type: none"> <li>1. (S) 시뮬레이션이 작동하고 있지 않다.</li> <li>2. (A) 사용자가 GUI에서 시뮬레이션을 시작한다.</li> <li>3. (S) 시스템의 시뮬레이션상태를 작동 중으로 바꾼다.</li> <li>4. (S) 승객 생성 설정에서 설정된 값에 따라 승객이 추가된다.</li> </ol>
Alternative Courses of Events	N/A
Exceptional Courses of Events	만약 시뮬레이션이 작동하고 있다면, 이 유즈케이스는 동작하지 않는다.

Use Case	2. 시뮬레이션 정지
Actor	User
Purpose	전체 시뮬레이션이 작동 중이면 정지한다.
Overview	사용자가 시뮬레이션을 작동 시킨다.
Type	Secondary
Cross Reference	Functions: R1.1, R1.2, R3.5

Pre-Requisites	시뮬레이션이 시작되어 있어야 한다.
Typical Courses of Events	(A) : Actor, (S) : System 1. (S) 시뮬레이션이 작동 하고 있다. 2. (A) 사용자가 GUI에서 시뮬레이션을 정지한다. 3. (S) 시스템의 시뮬레이션 상태를 멈춤으로 바꾼다. 4. (S) 승객 생성을 멈춘다.
Alternative Courses of Events	N/A
Exceptional Courses of Events	만약 시뮬레이션이 작동하고 있지 않다면 이 유즈케이스는 동작하지 않는다.

Use Case	3. 승객 생성 설정
Actor	User
Purpose	엘리베이터의 승객 수용량은 건물에 따라 다르므로 사용자가 설정하도록 한다.
Overview	단위시간당 몇 명의 승객을 생성할 지 정한다. 실제 단위로 입력받는다.
Type	Primary
Cross Reference	System functions : R3.5, R3.4 Use Case : 5.전체 승객 수 설정
Pre-Requisites	입력 값이 전체 승객 수를 넘지 않아야 한다.
Typical Courses of Events	(A) : Actor, (S) : System 1. (A) 사용자가 GUI에서 값을 설정한다. 2. (S) 전체 승객 수와 비교한다. 3. (S) 시뮬레이션에 바로 반영된다.
Alternative Courses of Events	임의로 지정된 값을 사용
Exceptional Courses of Events	Line 1:음수나 정수가 아닌 입력의 경우 에러를 발생시킨다. Line 2:입력된 값이 전체 승객 수보다 크면 전체 승객 수로 강제 조절하고 에러를 발생시킨다.

Use Case	4. 브레이크 강도 설정
Actor	User
Purpose	추락 상황시 자동으로 작동하는 브레이크의 강도를 설정한다.

Overview	추락 상황시에 브레이크가 자동으로 작동하게 되고 사용자는 브레이크가 작동 할 시 그에 대한 강도를 실제 단위로 설정 할 수 있다.(단위: N)
Type	Primary
Cross Reference	Functions: R4.3, R4.4, R3.2
Pre-Requisites	N/A
Typical Courses of Events	(A) : Actor, (S) : System 1. (A) 사용자가 GUI에서 브레이크의 강도를 설정한다. 2. (S) 사용자가 설정한 강도를 브레이크에 적용한다. 3. (S) 엘리베이터가 추락 시 설정한 강도에 맞게 브레이크가 작동한다.
Alternative Courses of Events	프로그램이 처음 시작하게 되면 디폴트로 지정된 값을 브레이크의 강도로 설정한다.
Exceptional Courses of Events	Line 1: 사용자가 0또는 음의 실수, NaN, Infinity로 설정 했다면, 에러를 내보내고 진행하지 않는다.

Use Case	5. 전체 승객수 설정
Actor	User
Purpose	생성된 승객의 전체 수의 제한을 사용자가 설정할 수 있도록 한다.
Overview	매 승객 생성 주기마다 현재 승객수와 추가될 승객 수를 확인 및 합산하여 설정된 전체 승객 수보다 크면, 그 차이만큼만 생성한다.
Type	Secondary
Cross Reference	System functions : R3.4 Use Case : 3.승객생성 설정
Pre-Requisites	N/A
Typical Courses of Events	(A) : Actor, (S) : System 1. (A) 사용자가 전체 승객수를 입력한다. 2. (S) 승객 생성 설정 값을 확인한다. 3. (S) 입력 값을 저장한다.
Alternative Courses of Events	미리 지정된 값으로 고정한다.
Exceptional Courses of Events	Line 1:음수나 정수가 아닌 입력의 경우 에러를 발생시킨다.

	Line 2:승객 생성 설정 값이 전체 승객 수보다 크면 강제로 일치되도록 낮춘다.
--	--

Use Case	6. 엘리베이터 무게 설정
Actor	User
Purpose	캐빈의 자체 무게를 설정한다.
Overview	사용자가 캐빈 자체의 무게를 설정 할 수 있다(단위: kg). 이렇게 설정된 값은 실제 물리연산에 적용된다. 탑승 승객의 무게와 함께 적용된다.
Type	Primary
Cross Reference	System functions : R4.4, R4.5 Use Case : 10.승객 무게 설정 13.캐빈 한계무게 설정
Pre-Requisites	N/A
Typical Courses of Events	(A) : Actor, (S) : System <ol style="list-style-type: none"> <li>1. (A) 사용자가 GUI에서 캐빈의 무게를 설정 한다.</li> <li>2. (S) 두개의 캐빈의 무게를 설정된 캐빈의 무게로 변경한다.</li> <li>3. (S) 이후 다른 연산에 변경된 캐빈의 무게가 적용된다.</li> </ol>
Alternative Courses of Events	프로그램이 처음 시작하게 되면 디폴트로 지정된 값을 캐빈의 무게로 설정한다.
Exceptional Courses of Events	Line 1 : 만약 사용자가 캐빈의 무게를 0 또는 음의 실수, NaN, Infinity로 설정한다면, 에러를 내보낸다.

Use Case	7. 중력 설정
Actor	User
Purpose	사용자가 중력을 임의로 설정할 수 있도록 한다.
Overview	입력된 중력값이 적당한 값인지 확인한 후 해당 값을 물리연산에 적용한다.
Type	Secondary
Cross Reference	System functions : R4.2
Pre-Requisites	N/A

Typical Courses of Events	(A) : Actor, (S) : System 1. (A) 사용자가 중력 값을 입력한다. 2. (S) 적절한 범위의 중력 값인지 확인한다. 3. (S) 물리연산에 입력값을 적용한다.
Alternative Courses of Events	지구중력으로 고정한다.
Exceptional Courses of Events	Line 1: 입력이 수가 아니라면 기본값으로 강제로 변경하고 오류를 출력한다. Line 2: 0초과 50이하의 실수값이 아니라면 범위 내로 강제 변경하고 오류를 출력한다.

Use Case	8. 모터 출력 설정
Actor	User
Purpose	캐빈을 작동시키는 모터의 출력을 설정한다.
Overview	캐빈은 모터의 힘으로 작동 되게 된다. 이때 탑승객과 캐빈 자체의 무게가 영향을 미치게 된다. 사용자는 여기에 사용 되는 모터의 출력을 설정할 수 있다.
Type	Primary
Cross Reference	System functions : R4.2, R3.2, R4.4 Use Case :
Pre-Requisites	N/A
Typical Courses of Events	(A) : Actor, (S) : System 1. (A) 사용자가 GUI에서 모터의 출력을 설정한다. 2. (S) 설정된 모터의 출력을 두개 캐빈의 모터에 반영한다. 3. (S) 이후 변경된 모터의 출력에 따라 캐빈의 움직임이 달라진다.
Alternative Courses of Events	프로그램이 처음 시작하게 되면 디폴트로 지정된 값을 모터의 출력으로 설정한다.
Exceptional Courses of Events	Line 1 : 사용자가 GUI에서 음의 실수 또는 NaN, Infinity를 설정 할 경우 에러를 내보낸다.

Use Case	9. 장애상황 수동 발생
Actor	User
Purpose	사용자가 장애 상황을 임의로 발생시킨다.

Overview	각 장애에 대해 GUI에서 4개의 버튼이 존재 하며 각 버튼을 누르면 해당 장애가 발생하게 된다. 하나의 장애가 발생 중이라면 다른 장애버튼은 비활성화 된다.
Type	Primary
Cross Reference	System functions : R2, R4.5, R4.4, R1.1, R1.2 Use Case :
Pre-Requisites	장애상황이 아니다.
Typical Courses of Events	(A) : Actor, (S) : System 1. (A) 사용자가 GUI에서 버튼을 눌러 장애를 발생시킨다. 2. (S) 다이얼로그를 띄워 각 장애상황에 맞는 정보를 입력하도록 한다. ○ 화재 : 층 ○ 추락 : 캐빈 번호 3. (S) 사용자가 발생 시킨 버튼에 해당 하는 장애상황의 상태로 시스템을 변경한다. 4. (S) 발생한 장애 발생 버튼은 눌림 상태로 표시를하고 다른 장애 발생 버튼은 비활성화 시킨다. 5. (S) 시스템이 장애에 대응한다. 6. (S) 시간이 지나며 시스템이 장애를 전부 처리했다면, 발생한 장애 발생 버튼과 함께 다른 장애 발생버튼들도 정상 상태로 활성화시킨다.
Alternative Courses of Events	캐빈의 승객이 인원수에 상관없이 캐빈의 한계 무게를 넘을 경우 해당 캐빈의 추락 장애 상황이 자동으로 발생된다.
Exceptional Courses of Events	N/A

Use Case	10. 승객 무게 설정
Actor	User
Purpose	사용자가 승객의 몸무게를 설정한다.
Overview	캐빈 초과 탑승여부를 판정할 시에 설정된 몸무게 값을 인원수에 곱하여 이용한다.
Type	Primary
Cross Reference	System functions : R3.2, R3.3, R4.5 Use Case : 11.정원 초과시 승객이 더 탈 확률 설정, 13.캐빈의 한계 무게 설정
Pre-Requisites	N/A

Typical Courses of Events	(A) : Actor, (S) : System 1. (A) 사용자가 승객 몸무게를 설정한다. 2. (S) 캐빈의 한계 무게와 캐빈 무게를 더하여 비교한다. 3. (S) 시뮬레이션에 반영한다. 4. (S) 정원이 초과될 시에 탑승객 인원내 몸무게 값을 곱하고 엘리베이터 무게에 더하여 캐빈 한계 무게와 비교한다. 한계를 초과한 경우 캐빈을 추락시킨다.
Alternative Courses of Events	성인 평균 몸무게로 미리 지정한다.
Exceptional Courses of Events	Line 1: 사용자의 입력이 수가 아니라면 강제로 기본값으로 변경하고 오류를 출력한다. Line 2: 캐빈의 한계 무게+캐빈 무게보다 크다면 강제로 동일하도록 변경하고 오류를 출력한다.

Use Case	11. 정원 초과시 승객이 더 탈 확률 설정
Actor	User
Purpose	정원 초과 상태에서 대기중인 승객이 있는 경우 더 탈 확률을 설정한다.
Overview	정원이 초과되어도 대기중인 승객이 무시하고 더 탈려고 하는 확률이다. 이에 따라 캐빈의 한계무게를 넘어서면 추락하게 된다.
Type	Primary
Cross Reference	System functions : R4.5, R4.4, R2 Use Case :
Pre-Requisites	N/A
Typical Courses of Events	(A) : Actor, (S) : System 1. (A) 사용자가 GUI 정원 초과시 대기중인 승객이 더 탈 확률을 설정한다. 2. (S) 각 층과 승객에 해당 확률을 반영한다. 3. (S) 이후 해당 확률대로 동작한다.
Alternative Courses of Events	N/A
Exceptional Courses of Events	Line 1: 만약 사용자가 0 초과 1이하의 값을 설정하지 않았다면 에러를 내보낸다.

Use Case	12. 캐빈 정원 설정
Actor	User

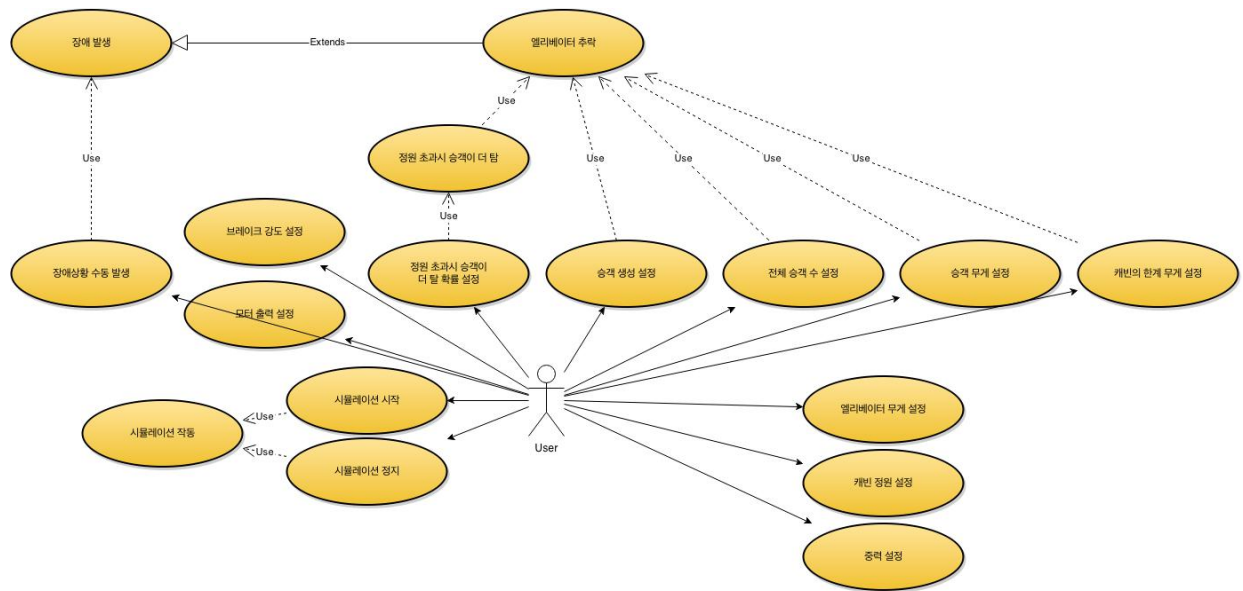
Purpose	캐빈의 정원을 사용자가 설정할 수 있도록 한다.
Overview	설정값보다 초과된 승객이 탑승하는 경우 초과경보를 올리고 층의 대기인원을 이용해 초과 탑승 사건을 확률 계산한다.
Type	Secondary
Cross Reference	System functions : R3.1, R3.2, R3.3, R4.5 Use Case : 10.승객 무게 설정 11.정원 초과시 승객이 더 탈 확률 설정 13.캐빈의 한계 무게 설정
Pre-Requisites	N/A
Typical Courses of Events	(A) : Actor, (S) : System 1. (A) 사용자가 캐빈의 정원을 설정한다. 2. (S) 입력값에 승객 무게를 곱하고 캐빈 무게와 합산하여 캐빈 한계 무게와 비교한다. 3. (S) 입력값을 시뮬레이션에 적용한다. 4. (S) 승객이 정원을 초과하여 탑승하는 경우 초과 경보를 올린다. 5. (S) 층에 대기중인 승객이 있다면 더 탈 확률 설정에 의해 더 타게하고 이 때에 한계 무게를 초과한다면 추락한다.
Alternative Courses of Events	미리 확률을 지정해둔다.
Exceptional Courses of Events	Line 1: 입력이 숫자가 아니거나 양의 정수가 아니면 강제로 기본값으로 변경하고 오류를 출력한다. Line 2: 캐빈 한계 무게를 초과하면 강제로 조정하고 오류를 출력한다.

Use Case	13. 캐빈 한계 무게 설정
Actor	User
Purpose	캐빈의 한계 무게를 사용자가 설정할 수 있게 한다.
Overview	한계 무게가 적절한 값인지 확인한 후 시뮬레이션에 적용하여 캐빈 추락조건에 이용한다.
Type	Secondary
Cross Reference	System functions : R4.5 R3.1 R3.2 R4.4 Use Case : 6.엘리베이터 무게 설정, 10.승객 무게 설정, 12.캐빈 정원 설정
Pre-Requisites	N/A

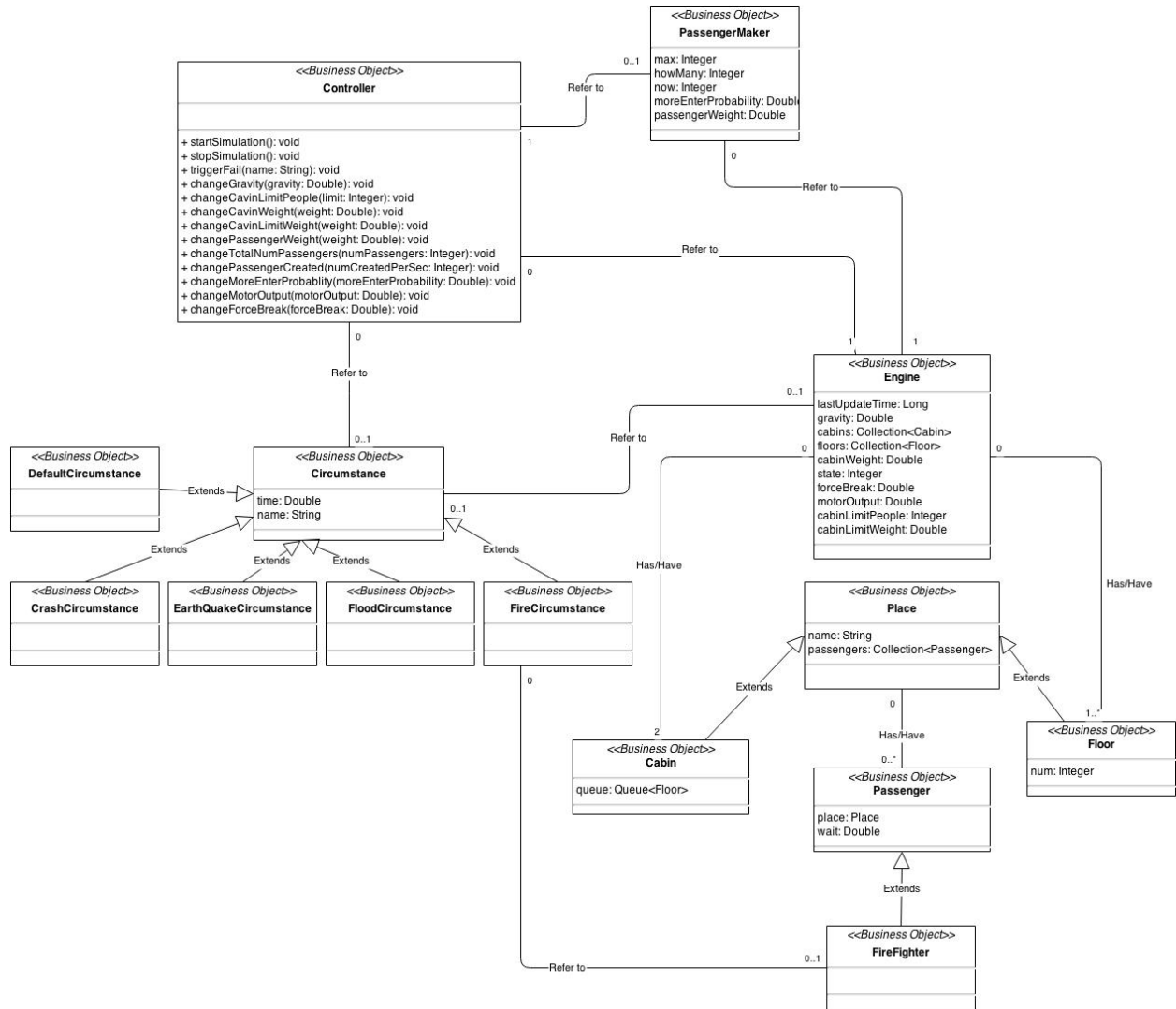


<p>Typical Courses of Events</p>	<p>(A) : Actor, (S) : System  1. (A) 사용자가 캐빈의 한계 무게를 설정한다.  2. (S) 승객 무게와 최대 탑승 인원을 곱하고 엘리베이터 무게와 합산하여 입력값과 비교한다.  3. (S) 시뮬레이션에 적용한다.  4. (S) 탑승인원의 전체 무게와 엘리베이터 무게가 캐빈의 한계무게를 초과하면 캐빈을 추락시킨다.</p>
<p>Alternative Courses of Events</p>	<p>미리 한계 무게를 지정해둔다.</p>
<p>Exceptional Courses of Events</p>	<p>Line 1: 사용자의 입력이 0을 초과하는 정수가 아니면 강제로 기본값으로 바꾸고 오류를 출력한다.  Line 2: 입력값이 최대정원 탑승시보다 작다면 일치하도록 강제로 변경하고 오류를 출력한다.</p>

### Activity 2132. Refine Use Case Diagrams



## Activity 2133. Define Domain Model



## Activity 2134. Refine Glossary

Term	Category	Comments
Controller	Class	사용자가 시뮬레이션 변수를 수정할 수 있는 인터페이스를 제공하고 변경된 값을 각 시뮬레이터의 클래스로 전달한다.
Circumstance	Class	각 장애상황 및 기본상황의 최상위 추상 클래스이다. 캐빈의 상황에 맞는 행동 트리거를 제공한다.

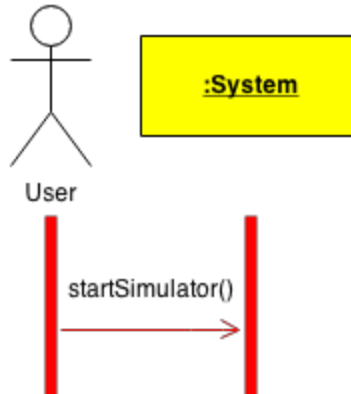
FireCircumstance	Class	화재 장애 상황일 때의 로직이 담겨있는 트리거를 제공한다.
FloodCircumstance	Class	수해 장애 상황일 때의 로직이 담겨있는 트리거를 제공한다.
EarthquakeCircumstance	Class	지진 장애 상황일 때의 로직이 담겨있는 트리거를 제공한다.
CrashCircumstance	Class	추락 장애 상황일 때의 로직이 담겨있는 트리거를 제공한다.
DefaultCircumstance	Class	장애 상황이 아닐 때의 로직이 담겨있는 트리거를 제공한다.
PassengerMaker	Class	컨트롤러의 설정값을 참조하여 정해진 주기마다 정해진 승객을 조건에 맞게 생성한다.
Engine	Class	컨트롤러의 설정된 물리값을 기반으로 캐빈의 작동을 물리적으로 묘사하고 UI에 맞도록 비율변환한다.
Passenger	Class	가상적으로 생성된 승객이다. 캐빈이나 층 중 한 곳을 필수적으로 위치값으로 가진다.
FireFighter	Class	화재 장애가 발생 했을 경우 화재에 대응하기 위한 클래스이다. 화재가 발생하면 1층에 추가된다.
Place	Class	승객이 위치로 필수적으로 한 개 가지고 있는 캐빈과 층의 상위 추상 클래스이다.
Cabin	Class	가상의 엘리베이터 객실이다. 현재 상황에 맞는 트리거대로 행동한다.
Floor	Class	가상의 층이다.
Circumstance.name	attribute	상황의 이름이다. 유저에게 상황을 알릴 때에 사용한다.
Circumstance.time	attribute	상황의 남은 제한시간
PassengerMaker.max	attribute	전체 승객 수 제한값
PassengerMaker.howMany	attribute	매 초 생성할 승객 수
PassengerMaker.now	attribute	현재 전체 승객 수

PassengerMaker.moreEnterProbability	attribute	캐빈이 정원 초과되어도 더 탈 확률
PassengerMaker.passengerWeight	attribute	승객 무게
Engine.gravity	attribute	중력값
Engine.cabins	attribute	두 개의 캐빈 콜렉션
Engine.floors	attribute	층 콜렉션
Engine.lastUpdateTime	attribute	엔진이 마지막으로 작동한 시각
Engine.state	attribute	프로그램 시뮬레이션 엔진의 현재 상태이다(0x00: STOP, 0x01: START, 0x02: FIRE, 0x04: CRASH, 0x08: FLOOD, 0x10: EARTHQUAKE, NORMAL: 0x20). 복합적으로 사용 할 수 있다.
Engine.forceBreak	attribute	브레이크의 강도(단위: N)
Engine.motorOutput	attribute	모터 출력(단위: N)
Engine.cabinWeight	attribute	캐빈의 자체무게
Engine.cabinLimitWeight	attribute	캐빈의 한계 무게
Engine.cabinLimitPeople	attribute	캐빈의 탑승 승객 한계 정원
Passenger.place	attribute	null일 수 없는 승객의 위치이다. Place 클래스가 지정된다.
Passenger.wait	attribute	승객을 현재 상태로 대기시킬 시간이다. (단위: ms)
Place.name	attribute	장소의 이름. 사용자에게 위치를 나타낼 때에 사용한다.
Place.passengers	attribute	현재 이 장소에 있는 승객의 콜렉션이다.
Cabin.queue	attribute	캐빈의 층 예약. Floor 클래스의 queue 자료구조형태.
Floor.num	attribute	층번호. 캐빈의 위치 표현에 내부적으로 사용된다.

## Activity 2135. Define System Sequence Diagrams

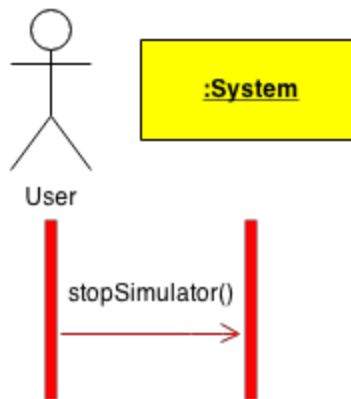
### USE CASE : 1.시뮬레이션 시작

- 1.시뮬레이션이 작동하고 있지 않다.
- 2.사용자가 GUI에서 시뮬레이션을 시작한다.
- 3.시스템의 시뮬레이션상태를 작동 중으로 바꾼다.
- 4.승객 생성 설정에서 설정된 값에 따라 승객이 추가된다.



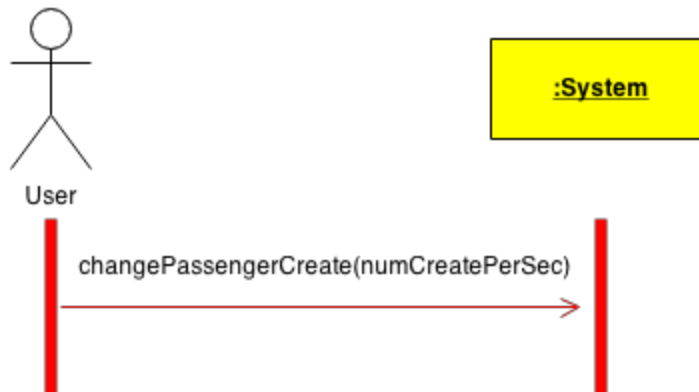
### USE CASE : 2.시뮬레이션 정지

1. 시뮬레이션이 작동 하고 있다.
2. 사용자가 GUI에서 시뮬레이션을 정지한다.
3. 시스템의 시뮬레이션 상태를 멈춤으로 바꾼다.
4. 승객 생성을 멈춘다.



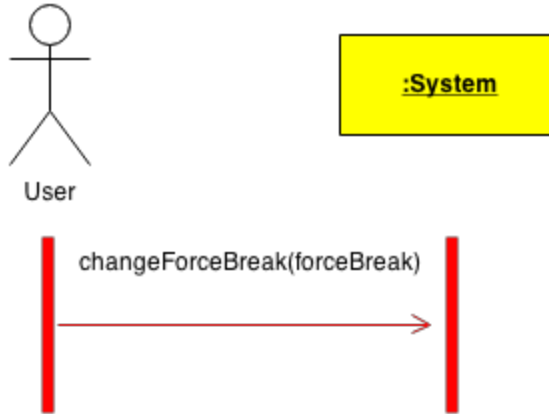
### USE CASE : 3.승객 생성 설정

1. 사용자가 GUI에서 값을 설정한다.
2. 전체 승객 수와 비교한다.
3. 시뮬레이션에 바로 반영된다.



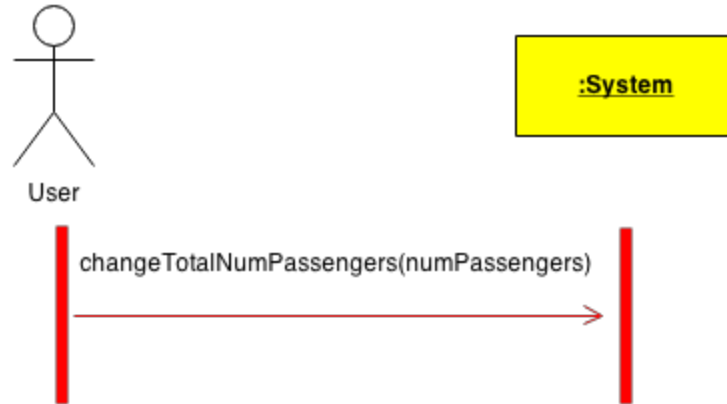
**USE CASE : 4. 브레이크 강도 설정**

1. 사용자가 GUI에서 브레이크의 강도를 설정한다.
2. 사용자가 설정한 강도를 브레이크에 적용한다.
3. 엘리베이터가 추락 시 설정한 강도에 맞게 브레이크가 작동한다.



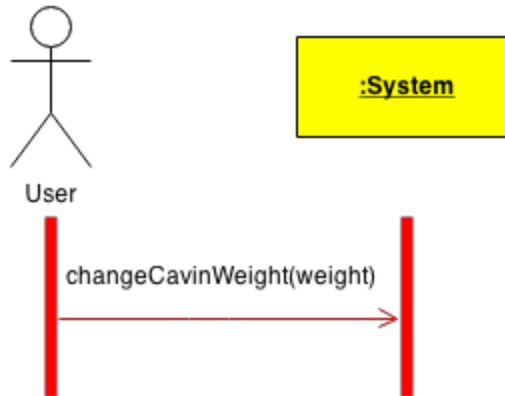
**USE CASE : 5. 전체 승객수 설정**

1. 사용자가 전체 승객수를 입력한다.
2. 승객 생성 설정 값을 확인한다.



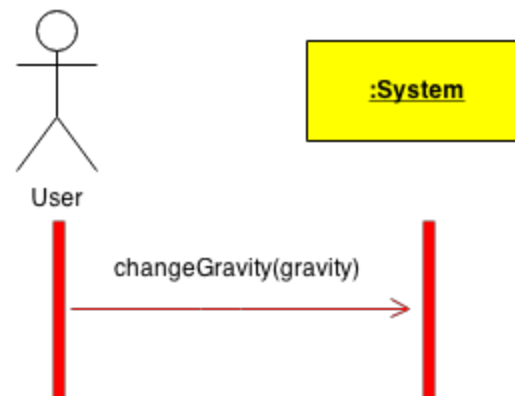
**USE CASE : 6. 엘리베이터 무게 설정**

1. 사용자가 GUI에서 캐빈의 무게를 설정 한다.
2. 두개의 캐빈의 무게를 설정된 캐빈의 무게로 변경한다.
3. 이후 다른 연산에 변경된 캐빈의 무게가 적용된다.



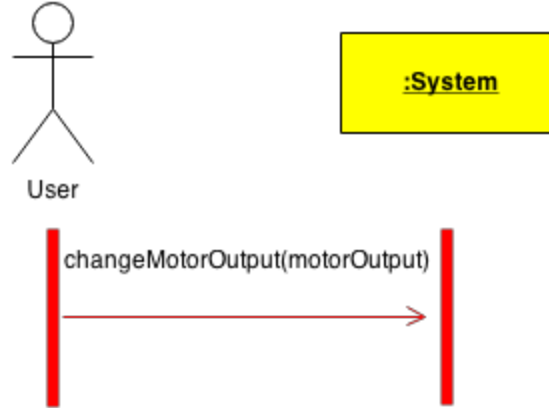
**USE CASE : 7. 중력 설정**

1. 사용자가 중력 값을 입력한다.
2. 적절한 범위의 중력 값인지 확인한다.
3. 물리엔진에 입력값을 적용한다.



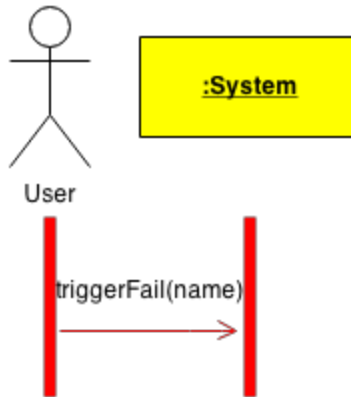
**USE CASE : 8.모터 출력 설정**

1. 사용자가 GUI에서 모터의 출력을 설정한다.
2. 설정된 모터의 출력을 두개 캐빈의 모터에 반영한다.
3. 이후 변경된 모터의 출력에 따라 캐빈의 움직임이 달라진다.



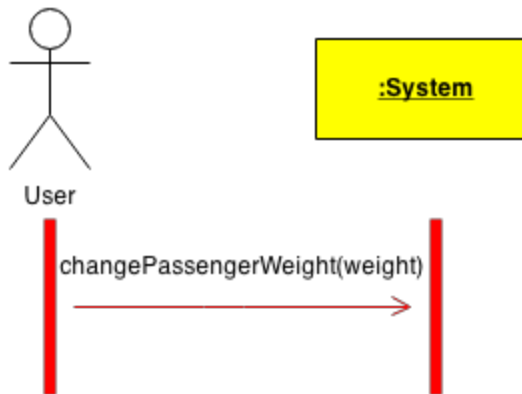
**USE CASE : 9.장애상황 수동 발생**

1. 사용자가 GUI에서 버튼을 눌러 장애를 발생시킨다.
2. 다이얼로그를 띄워 각 장애상황에 맞는 정보를 입력하도록 한다.  
화재 : 층  
추락 : 캐빈 번호
3. 사용자가 발생 시킨 버튼에 해당하는 장애상황의 상태로 시스템을 변경한다.
4. 발생한 장애 발생 버튼은 눌림 상태로 표시를하고 다른 장애 발생 버튼은 비활성화 시킨다.
5. 시스템이 장애에 대응한다.
6. 시간이 지나며 시스템이 장애를 전부 처리했다면, 발생한 장애 발생 버튼과 함께 다른 장애 발생버튼들도 정상 상태로 활성화시킨다.



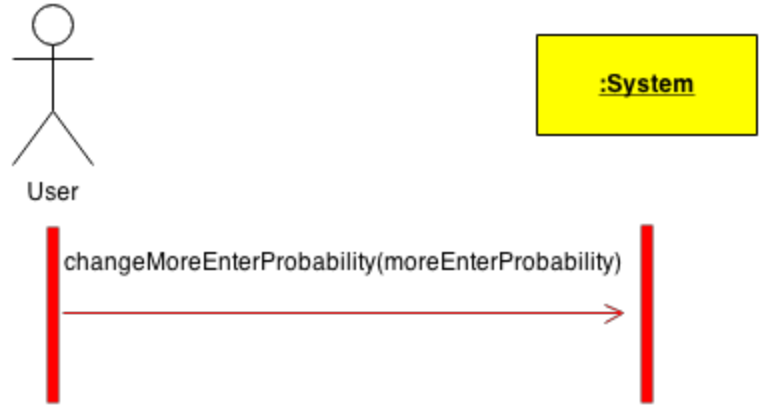
**USE CASE : 10.승객 무게 설정**

1. 사용자가 승객 몸무게를 설정한다.
2. 캐빈의 한계 무게와 캐빈 무게를 더하여 비교한다.
3. 시뮬레이션에 반영한다.
4. 정원이 초과될 시에 탑승객 인원내 몸무게 값을 곱하고 엘리베이터 무게에 더하여 캐빈 한계 무게와 비교한다. 한계를 초과한 경우 캐빈을 추락시킨다.



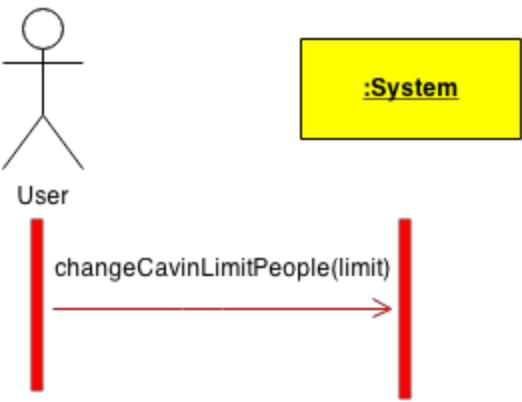
**USE CASE : 11.정원 초과시 승객이 더 탈 확률 설정**

1. 사용자가 GUI 정원 초과시 대기중인 승객이 더 탈 확률을 설정한다.
2. 각 층과 승객에 해당 확률을 반영한다.
3. 이후 해당 확률대로 동작한다.



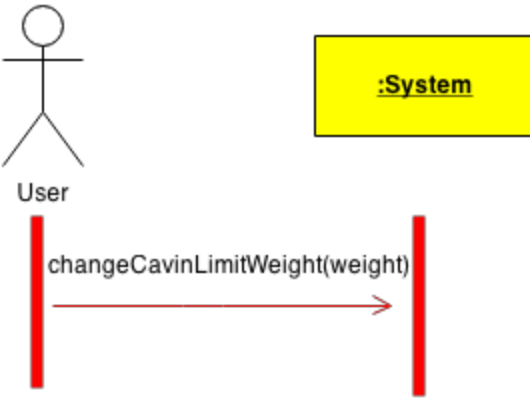
**USE CASE : 12.캐빈 정원 설정**

1. 사용자가 캐빈의 정원을 설정한다.
2. 입력값에 승객 무게를 곱하고 캐빈 무게와 합산하여 캐빈 한계 무게와 비교한다.
3. 입력값을 시뮬레이션에 적용한다.
4. 승객이 정원을 초과하여 탑승하는 경우 초과 경보를 울린다.
5. 층에 대기중인 승객이 있다면 더 탈 확률 설정에 의해 더 타게하고 이 때에 한계 무게를 초과한다면 추락한다.



**USE CASE : 13.캐빈 한계 무게 설정**

1. 사용자가 캐빈의 한계 무게를 설정한다.
2. 승객 무게와 탑승 인원을 곱하고 엘리베이터 무게와 합산하여 입력값과 비교한다.
3. 시뮬레이션에 적용한다.
4. 탑승인원의 전체 무게와 엘리베이터 무게가 캐빈의 한계무게를 초과하면 캐빈을 추락시킨다.



**Activity 2136. Define Operation Contracts**

1. Use-case and System operations mapping table

Use-case	System operations
1. 시뮬레이션 시작	1. startSimulation()
2. 시뮬레이션 정지	2. stopSimulation()



3. 승객 생성 설정	3. changePassengerCreated()
4. 브레이크 강도 설정	4. changeForceBreak()
5. 전체 승객 수 설정	5. changeTotalNumPassengers()
6. 엘리베이터 무게 설정	6. changeCavinWeight()
7. 중력 설정	7. changeGravity()
8. 모터 출력 설정	8. changeMotorOutput()
9. 장애상황 수동 발생	9. triggerFail()
10. 승객 무게 설정	10. changePassengerWeight()
11. 정원 초과시 승객이 더 탈 확률 설정	11. changeMoreEnterProbability()
12. 캐빈 정원 설정	12. changeCavinLimitPeople()
13. 캐빈의 한계 무게 설정	13. changeCavinLimitWeight()

## 2. System operation descriptions

<b>Name</b>	1. startSimulation()
<b>Responsibilities</b>	만약 Engine.state가 STOP일 경우 simulation을 시작한다.
<b>Type</b>	Controller
<b>Cross References</b>	System functions: R1.1 Use-cases: 1. 시뮬레이션 시작
<b>Notes</b>	
<b>Exceptions</b>	만약 Engine.state가 START일 경우, 에러를 내보내고 아무 동작도 하지 않는다.
<b>Output</b>	N/A
<b>Pre-conditions</b>	Engine.state가 STOP이어야 한다.
<b>Post-conditions</b>	Engine을 GUI의 repaint 이벤트마다 업데이트 시킨다. Engine.state를 START로 바꾼다. PassengerMaker에서 Passenger를 매초 PassengerMaker.howMany만큼 랜덤하게 여러 층에 생성하기 시작한다.

<b>Name</b>	<b>2. stopSimulation()</b>
<b>Responsibilities</b>	만약 Engine.state가 START일 경우 simulation을 시작한다.
<b>Type</b>	Controller
<b>Cross References</b>	System functions: R1.2 Use-cases: 2. 시뮬레이션 정지
<b>Notes</b>	
<b>Exceptions</b>	만약 Engine.state가 STOP일 경우, 에러를 내보낸다.
<b>Output</b>	N/A
<b>Pre-conditions</b>	N/A
<b>Post-conditions</b>	Engine을 GUI의 repaint 이벤트마다 업데이트 시키지 않는다. Engine.state를 STOP로 바꾼다. PassengerMaker를 멈춘다.

<b>Name</b>	<b>3. changePassengerCreated()</b>
<b>Responsibilities</b>	매초마다 생성되는 승객의 수가 설정된 값으로 변경 된다.
<b>Type</b>	Controller
<b>Cross References</b>	System functions: R3.5
<b>Notes</b>	
<b>Exceptions</b>	만약 음수나 정수가 아닌 입력의 경우 에러를 발생시킨다. 만약 입력된 값이 전체 승객 수보다 크면 전체 승객 수로 강제 조절하고 에러를 발생시킨다.
<b>Output</b>	N/A
<b>Pre-conditions</b>	N/A
<b>Post-conditions</b>	PassengerMaker.howMany의 값이 바뀐다.

<b>Name</b>	<b>4. changeForceBreak()</b>
<b>Responsibilities</b>	캐빈 추락시 작동하는 브레이크의 강도가 설정된 값으로 변경된다.
<b>Type</b>	Controller

<b>Cross References</b>	System functions: R4.3
<b>Notes</b>	
<b>Exceptions</b>	만약 사용자가 0또는 음의 실수, NaN, Infinity로 설정 했다면, 에러를 내보내고 진행하지 않는다.
<b>Output</b>	N/A
<b>Pre-conditions</b>	N/A
<b>Post-conditions</b>	Engine.forceBreak의 값이 바뀐다.

<b>Name</b>	<b>5. changeTotalNumPassengers()</b>
<b>Responsibilities</b>	캐빈 추락시 작동하는 브레이크의 강도가 설정된 값으로 변경된다.
<b>Type</b>	Controller
<b>Cross References</b>	System functions: R3.4 Use-cases: 5. 전체 승객수 설정
<b>Notes</b>	
<b>Exceptions</b>	만약 음수나 정수가 아닌 입력의 경우 에러를 발생시킨다. 만약 승객 생성 설정 값이 전체 승객 수보다 크면 강제로 일치되도록 낮춘다.
<b>Output</b>	N/A
<b>Pre-conditions</b>	N/A
<b>Post-conditions</b>	PassengerMaker.max의 값이 변경된다. 값이 줄어들면 기존 승객은 즉시 사라지지 않고 동작한다. PassengerMaker.now는 새로 생성되는 Passenger에 한해서 PassengerMaker.max보다 클 수 없다.

<b>Name</b>	<b>6. changeCavinWeight()</b>
<b>Responsibilities</b>	캐빈 자체에 무게를 설정된 값으로 변경한다.
<b>Type</b>	Controller
<b>Cross References</b>	System functions: R4.4 Use-cases: 6. 엘리베이터 무게 설정

<b>Notes</b>	
<b>Exceptions</b>	만약 사용자가 캐빈의 무게를 0 또는 음의 실수, NaN, Infinity로 설정한다면, 에러를 내보낸다.
<b>Output</b>	N/A
<b>Pre-conditions</b>	N/A
<b>Post-conditions</b>	Engine.cabinWeight의 값이 변경된다.

<b>Name</b>	<b>7. changeGravity()</b>
<b>Responsibilities</b>	캐빈에 미치고 있는 중력을 설정된 값으로 변경한다.
<b>Type</b>	Controller
<b>Cross References</b>	System functions: R4.2 Use-cases: 7. 중력 설정
<b>Notes</b>	
<b>Exceptions</b>	만약 입력이 수가 아니라면 기본값으로 강제로 변경하고 오류를 출력한다. 만약 0초과 50이하의 실수값이 아니라면 범위 내로 강제 변경하고 오류를 출력한다.
<b>Output</b>	N/A
<b>Pre-conditions</b>	N/A
<b>Post-conditions</b>	Engine.gravity의 값이 변경된다.

<b>Name</b>	<b>8. changeMotorOutput()</b>
<b>Responsibilities</b>	캐빈을 작동시키는 모터의 출력을 설정된 값으로 변경한다.
<b>Type</b>	Controller
<b>Cross References</b>	System functions: R4.2, R3.2, R4.4 Use-cases: 8. 모터 출력 설정
<b>Notes</b>	
<b>Exceptions</b>	만약 사용자가 GUI에서 음의 실수 또는 NaN, Infinity를 설정 할 경우 에러를 내보낸다.
<b>Output</b>	N/A

<b>Pre-conditions</b>	N/A
<b>Post-conditions</b>	Engine.motorOutput의 값이 변경된다.

<b>Name</b>	<b>9. triggerFail()</b>
<b>Responsibilities</b>	사용자가 장애 상황을 수동으로 발생시킨다.
<b>Type</b>	Controller
<b>Cross References</b>	System functions: R2, R4.5, R4.4, R1.1, R1.2 Use-cases: 9. 장애상황 수동 발생
<b>Notes</b>	상황 클래스인 Circumstance는 strategy패턴을 사용하여 DefaultCircumstance, FireCircumstance, FloodCircumstance, CrashCircumstance, EarthQuakeCircumstance중 한개가 되어 장애상황을 발생시키게 된다.
<b>Exceptions</b>	만약 사용자가 GUI에서 음의 실수 또는 NaN, Infinity를 설정 할 경우 에러를 내보낸다.
<b>Output</b>	N/A
<b>Pre-conditions</b>	Engine.state가 START&NORMAL이거나 START이어야 한다.
<b>Post-conditions</b>	<p>장애가 발생 되는 동안 PassengerMaker에서 Passenger가 생성되지 않는다. 활성화한 장애버튼 외에 다른 장애발생 버튼은 비활성화 된다. Circumstance의 전략을 해당 장애상황에 맞는 4개의 클래스중 하나로 바꾼다.</p> <ul style="list-style-type: none"> <li>● FireCircumstance <ul style="list-style-type: none"> <li>○ Engine.state가 START&amp;FIRE로 바뀐다.</li> <li>○ 화재가 난 층에 승객은 전부 죽는다.</li> <li>○ 1층에 FireFighter로 1층에 해당 하는 Floor에 추가된다.</li> <li>○ Cavin은 1층으로 가게 되고, FireFighter를 화재가 발생한 층(Floor)에 이동한다.</li> <li>○ FireFighter가 도착하면 화재 장애는 종료된다.</li> <li>○ 이 동안 다른 캐빈의 호출은 무시한다.</li> </ul> </li> <li>● FloodCircumstance <ul style="list-style-type: none"> <li>○ Engine.state가 START&amp;FLOOD로 바뀐다.</li> <li>○ 1층에 있는 승객은 전부 죽는다.</li> <li>○ 1분 동안 수해 장애상황은 지속된다.</li> <li>○ 1분 후 수해 장애는 종료된다.</li> <li>○ 이 동안 캐빈은 지상 10층에 멈춰있다.</li> </ul> </li> <li>● EarthQuakeCircumstance <ul style="list-style-type: none"> <li>○ Engine.state가 START&amp;EARTHQUAKE로 바뀐다.</li> <li>○ 1분 동안 지진 장애 상황이 지속된 후 종료된다.</li> </ul> </li> </ul>

	<ul style="list-style-type: none"> <li>○ 매초마다 전체 승객 중 랜덤하게 한명씩 죽는다.</li> <li>○ 캐빈의 움직임이 현재 층에서 멈춘다.</li> <li>● CrashCircumstance <ul style="list-style-type: none"> <li>○ Engine.state가 START&amp;CRASH로 바뀐다.</li> <li>○ 선택된 캐빈은 추락 하고 브레이크가 작동 하게 된다.</li> <li>○ Engine.forceBreak를 캐빈의 추락하는 힘에 빼고 지하 1층에 도착하는 충격량이 8333.333N*s 이상이면 전부 사망한다.</li> <li>○ 캐빈이 지하 1층에 도착하거나 브레이크에 의해 멈춘다면 추락 장애는 종료된다.</li> </ul> </li> </ul> <p>상황 종료 후에 Engine.state가 START&amp;NORMAL로 바뀐다.</p>
--	---

<b>Name</b>	<b>10. changePassengerWeight()</b>
<b>Responsibilities</b>	사용자가 승객의 몸무게를 설정한다.
<b>Type</b>	Controller
<b>Cross References</b>	System functions : R3.2, R3.3, R4.5 Use Case : 10.승객 무게 설정
<b>Notes</b>	캐빈 초과 탑승여부를 판정할 시에 설정된 몸무게 값을 인원에 곱하여 이용한다.
<b>Exceptions</b>	사용자의 입력이 수가 아니라면 강제로 기본값으로 변경하고 오류를 출력한다. 캐빈의 한계 무게+캐빈 무게보다 크다면 강제로 동일하도록 변경하고 오류를 출력한다.
<b>Output</b>	N/A
<b>Pre-conditions</b>	N/A
<b>Post-conditions</b>	PassengerMaker.passengerWeight 를 수정한다.

<b>Name</b>	<b>11. changeMoreEnterProbability()</b>
<b>Responsibilities</b>	정원 초과 상태에서 대기중인 승객이 더 탈 확률을 설정된 값으로 변경한다.
<b>Type</b>	Controller
<b>Cross References</b>	System functions : R4.5, R4.4, R2 Use-Cases: 11. 정원 초과시 승객이 더 탈
<b>Notes</b>	

<b>Exceptions</b>	만약 사용자가 0 초과 1이하의 값을 설정하지 않았다면 에러를 내보낸다.
<b>Output</b>	N/A
<b>Pre-conditions</b>	N/A
<b>Post-conditions</b>	PassengerMaker.moreEnterProbability의 값이 바뀐다.

<b>Name</b>	<b>12. changeCavinLimitPeople()</b>
<b>Responsibilities</b>	캐빈의 한계 정원을 설정된 값으로 변경한다.
<b>Type</b>	Controller
<b>Cross References</b>	System functions : R3.1, R3.2, R3.3, R4.5 Use Case : 10.승객 무게 설정, 11.정원 초과시 승객이 더 탈 확률 설정, 13. 캐빈의 한계 무게 설정, 12. 캐빈 정원 설정
<b>Notes</b>	
<b>Exceptions</b>	만약 입력이 숫자가 아니거나 양의 정수가 아니면 강제로 기본값으로 변경하고 오류를 출력한다. 만약 캐빈 한계 무게를 초과하면 강제로 조정하고 오류를 출력한다.
<b>Output</b>	N/A
<b>Pre-conditions</b>	N/A
<b>Post-conditions</b>	Engine.cabinLimitPeople의 값이 바뀐다.

<b>Name</b>	<b>13. changeCavinLimitWeight()</b>
<b>Responsibilities</b>	캐빈의 한계 무게를 사용자가 설정할 수 있게 한다.
<b>Type</b>	Controller
<b>Cross References</b>	System functions : R4.5 R3.1 R3.2 R4.4 Use Case : 13.캐빈 한계 무게 설정
<b>Notes</b>	한계 무게가 적절한 값인지 확인한 후 시뮬레이션에 적용하여 캐빈 추락조건에 이용한다.
<b>Exceptions</b>	사용자의 입력이 0을 초과하는 정수가 아니면 강제로 기본값으로 바꾸고 오류를 출력한다. 입력값이 최대정원 탑승시보다 작다면 일치하도록 강제로 변경하고 오류를 출력한다.

<b>Output</b>	N/A
<b>Pre-conditions</b>	N/A
<b>Post-conditions</b>	Engine.cabinLimitWeight 을 수정한다.