



Category Partition Testing Tool

201414134 오세욱

201414136 임현유

201211375 임동현

201211387 하현규

Index

1. **Revise Plan**
2. **Define UI**
3. **Define Real Use-Case**
4. **Define Interaction Diagrams**
5. **Define Design Class Diagrams**
6. **Design Traceability Analysis**

Revise Plan

1.1 Use Case

Revise Plan

Use Case
Category 추가
Category 수정
Category 삭제
대푯값 추가
대푯값 수정
대푯값 삭제
Constraint 추가
Constraint 수정
Constraint 삭제
대푯값 선택
Test Case 계산
출력
초기화
프로그램 종료



Use Case
Category 추가
Category 수정
Category 삭제
대푯값 추가
대푯값 수정
대푯값 삭제
Constraint 추가
Constraint 삭제
대푯값 선택
Test Case 계산
초기화

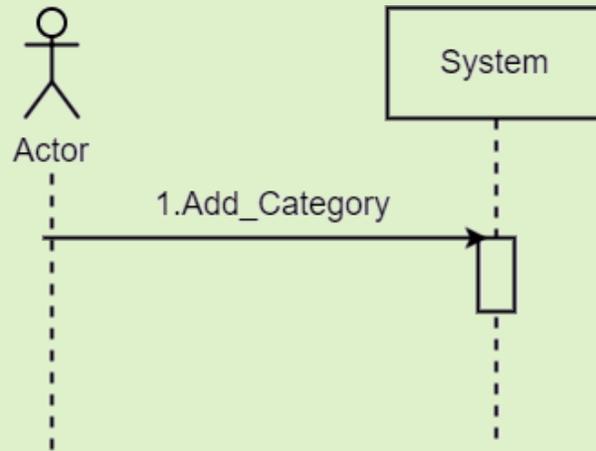
Constraint 수정 제거
출력 제거
프로그램 종료 제거

1.2 System Sequence Diagram

Revise Plan

Use Case : Category 추가

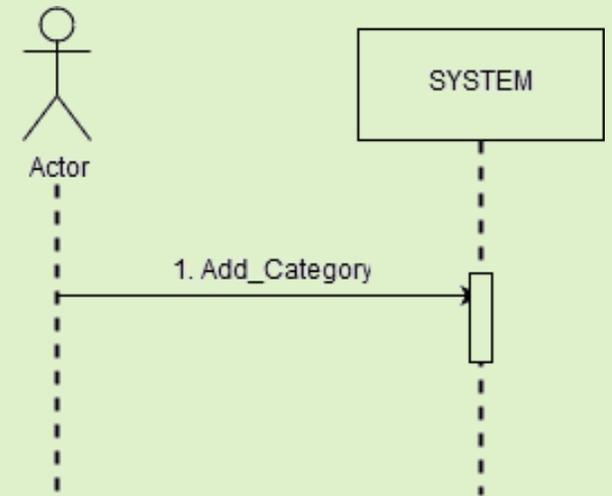
- 1) Category 추가를 요청한다.
- 2) Category의 이름을 입력한다.
- 3) 입력 받은 이름을 갖는 Category를 생성한다.



Category 추가
System Sequence Diagram에
별도의 이름 입력 없이 Category 생성

Use Case : Category 추가

- 1) Category 추가를 요청한다.
- 2) Category를 생성한다.

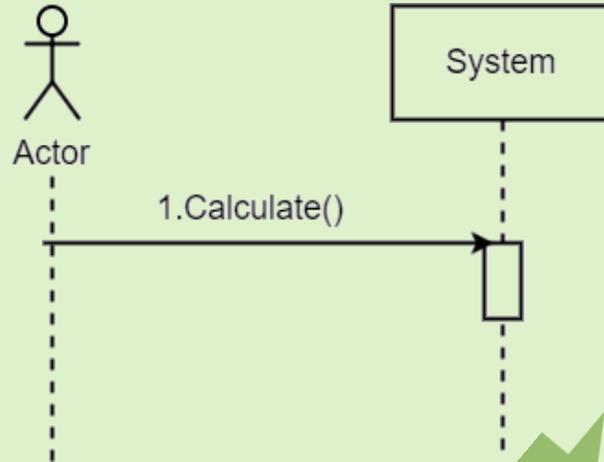


1.2 System Sequence Diagram

Revise Plan

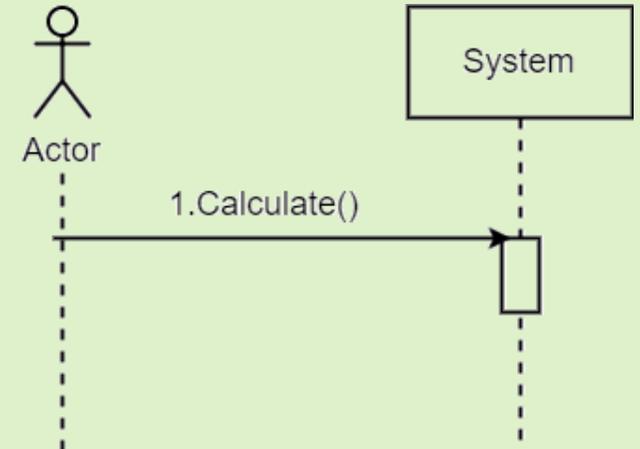
Use Case : Test Case 계산

- 1) Test Case 계산을 요청한다.
- 2) 입력된 값에 따라 Test Case를 계산한다.



Use Case : Test Case 계산

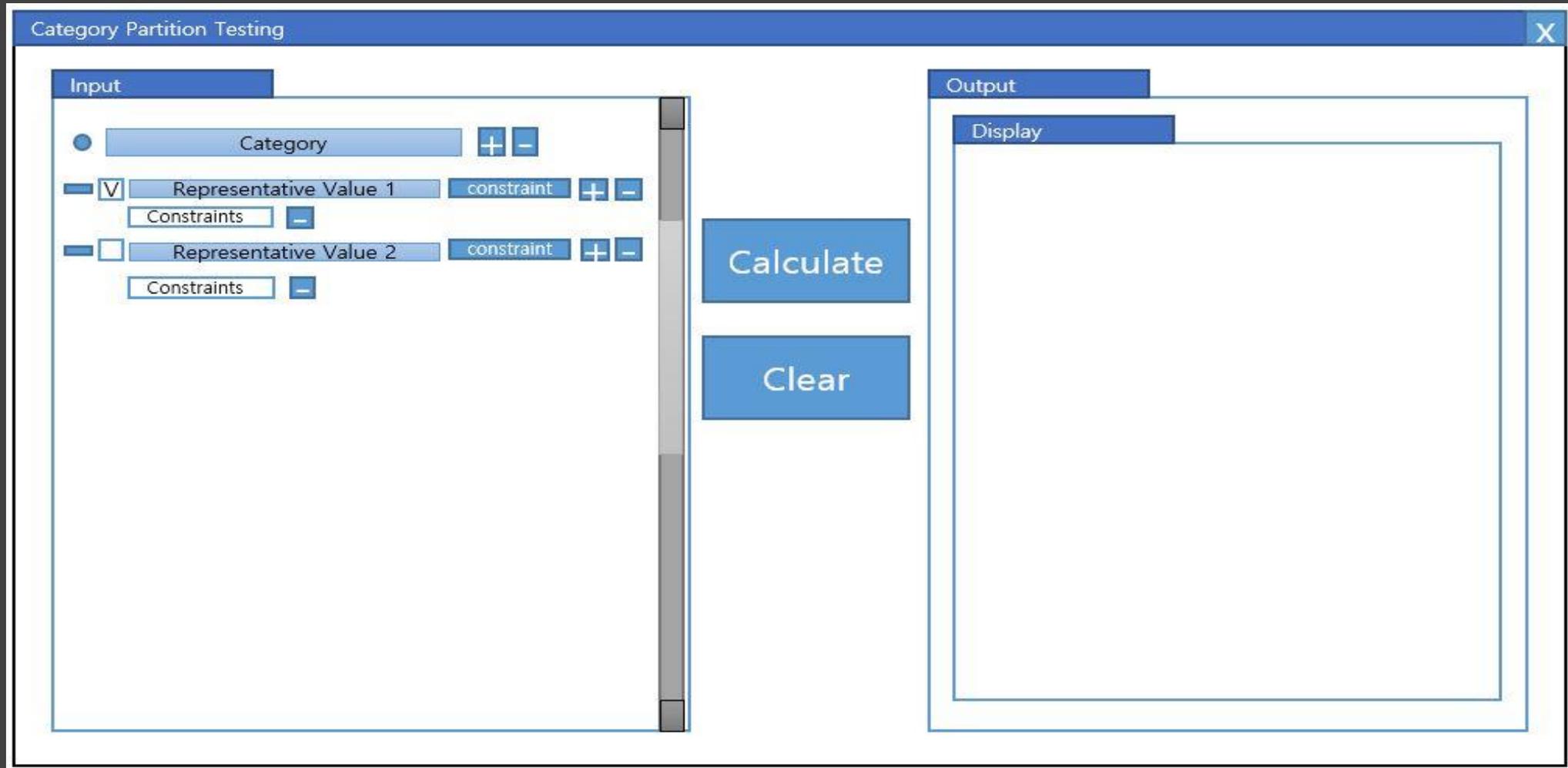
- 1) Test Case 계산을 요청한다.
- 2) Test Case 를 계산한다.
- 3) 계산된 결과를 출력한다.



Test Case System Sequence Diagram에 출력기능 포함

1.3 Implement Prototype

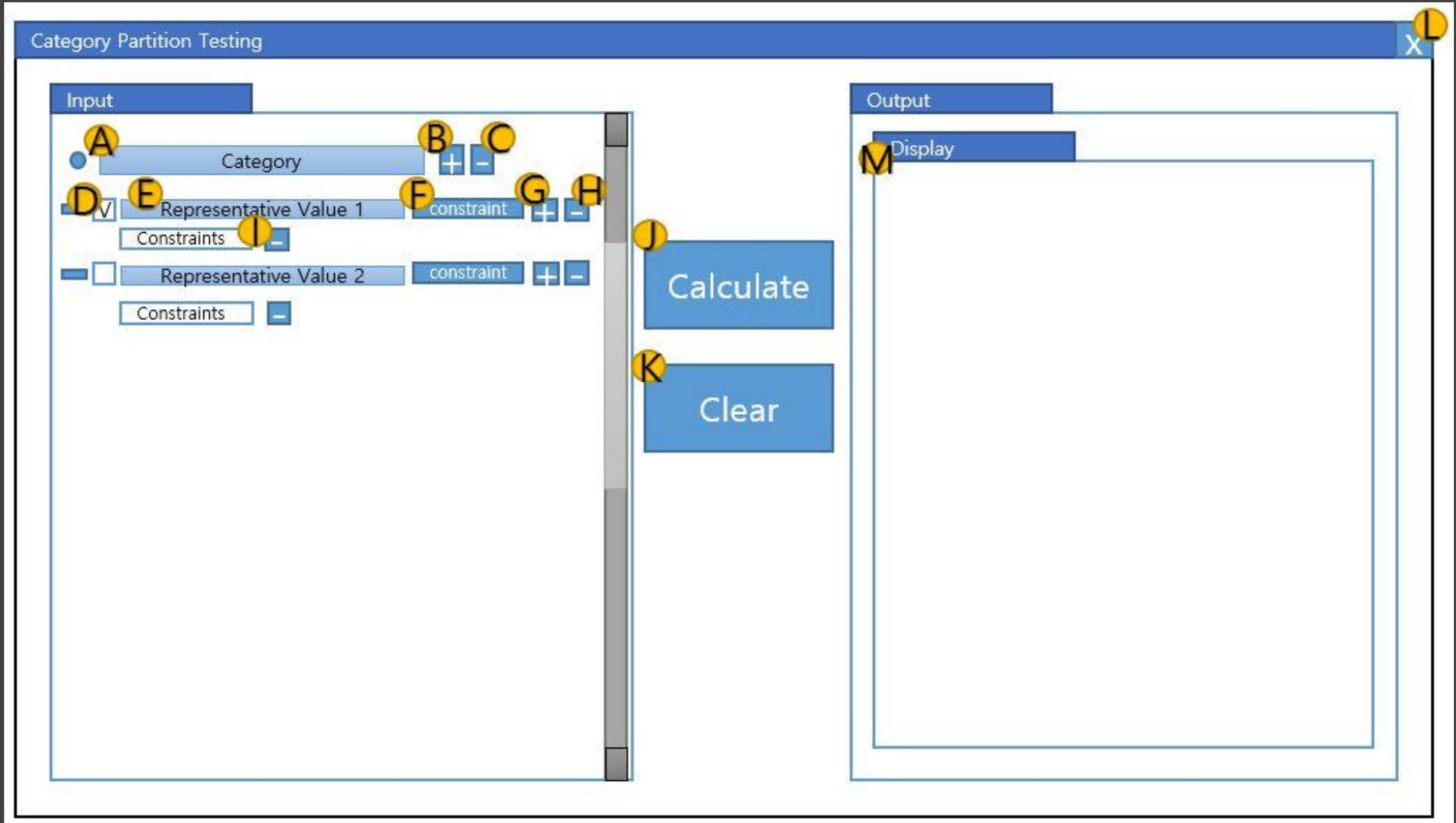
Revise Plan



Constraint버튼 수정

Define UI

Define UI



Define Real Use-Case

Define Real Use Cases(1/11)

Use Case	Category 추가
Actor	Actor-based
Purpose	Category를 추가한다.
Overview	User에게 Category 추가를 요청하면 Category를 추가한다..
Type	Primary
Cross Reference	Functions: R1.1 Use Cases:
Pre-Requisites	N/A
Typical Courses of Events	(A):Actor, (S):System 1. (A):Category 추가(B)버튼을 누른다. 2. (S):이름이 'Category #Category_Num'인 Category 객체를 생성한다.
Alternative Courses of Events	N/A
Exceptional Courses of Events	N/A

Define Real Use Cases(2/11)

Use Case	Category 수정
Actor	Actor-based
Purpose	Category를 수정한다.
Overview	User가 Category 객체의 이름을 수정한다.
Type	Primary
Cross Reference	Functions: R1.2 Use Cases:
Pre-Requisites	수정 할 Category 객체가 존재해야 한다.
Typical Courses of Events	(A):Actor, (S):System 1. (A):Category Name_Field(A)에 Category 이름을 입력한다. 2. (S):입력받은 Category_Id와 일치하는 Category 객체를 찾는다.. 3. (S):선택 된 Category 객체의 이름을 입력 받은 이름으로 수정한다.
Alternative Courses of Events	N/A
Exceptional Courses of Events	Category 이름으로 ""을 입력 할 경우 변경하지 않는다.

Define Real Use Cases(3/11)

Use Case	Category 삭제
Actor	Actor-based
Purpose	Category를 삭제한다.
Overview	User가 Category 삭제를 요청하면 Category 객체를 삭제한다.
Type	Primary
Cross Reference	Functions: R1.3 Use Cases:
Pre-Requisites	삭제 할 Category가 존재해야 한다.
Typical Courses of Events	(A):Actor, (S):System 1. (A):Category 삭제 버튼(C)를 누른다. 2. (S):Category 삭제 여부를 사용자에게 확인한다. 3. (A):Category 삭제를 승인한다. 4. (S):Category 객체를 삭제한다.
Alternative Courses of Events	N/A
Exceptional Courses of Events	1. Category 삭제를 승인하지 않을 경우 삭제를 취소한다. 2. Category 객체 내 Property를 가진 대포값이 존재하면 삭제되지 않는다.

Define Real Use Cases(4/11)

Use Case	대푯값 추가
Actor	Actor-based
Purpose	대푯값을 추가한다.
Overview	User가 대푯값 추가를 요청하면 RV 객체를 추가한다.
Type	Primary
Cross Reference	Functions: R1.4 Use Cases:
Pre-Requisites	Category 객체가 존재해야 한다.
Typical Courses of Events	(A):Actor, (S):System 1. (A):대푯값 생성 버튼(G)를 누른다. 2. (S):입력받은 Category_Id와 일치하는 Category 객체를 찾는다. 3. (S):'Representative Value#RV_Num'의 이름을 갖는 RV를 생성한다.
Alternative Courses of Events	N/A
Exceptional Courses of Events	N/A

Define Real Use Cases(5/11)

Use Case	대포깃 수정
Actor	Actor-based
Purpose	대포깃을 수정한다.
Overview	User가 대포깃 수정을 RV 객체의 이름을 수정한다.
Type	Primary
Cross Reference	Functions: R1.5 Use Cases:
Pre-Requisites	대포깃이 존재해야 한다.
Typical Courses of Events	(A):Actor, (S):System 1. (A):RV Name_Field(E)를 누른다. 2. (S):입력받은 Category_Id와 일치하는 Category 객체를 찾는다. 3. (S):'Representative Value#RV_Num'의 이름을 갖는 RV를 생성한다. 4. (S):선택 된 RV 객체의 이름을 입력 받은 이름으로 수정한다.
Alternative Courses of Events	N/A
Exceptional Courses of Events	RV 이름으로 ""을 입력 할 경우 변경하지 않는다.

Define Real Use Cases(6/11)

Use Case	대푯값 삭제
Actor	Actor-based
Purpose	대푯값을 삭제한다.
Overview	User가 대푯값 삭제를 요청하면 RV 객체를 삭제한다.
Type	Primary
Cross Reference	Functions: R1.6 Use Cases:
Pre-Requisites	대푯값이 존재해야 한다.
Typical Courses of Events	(A):Actor, (S):System 1. (A):RV 삭제 버튼(H)를 누른다. 2. (S):Category 삭제 여부를 사용자에게 확인한다. 3. (A):Category 삭제를 승인한다. 4. (S):입력 받은 Category_Id와 일치하는 Category 객체를 찾는다. 5. (S):입력 받은 RV_Id와 일치하는 RV 객체를 찾는다. 6. (S):해당 RV 객체의 Property_List를 가져온다. 7. (S):가져온 Property_List에 해당하는 모든 If Constraint를 삭제하고 Input_Manager.Property_List를 갱신한다. 8. RV 객체를 삭제한다.

Define Real Use Cases(6/11)

Use Case	대포깃 삭제
Alternative Courses of Events	N/A
Exceptional Courses of Events	User가 Representative Value 삭제를 승인하지 않으면 삭제를 취소한다.

Define Real Use Cases(7/11)

Use Case	Constraint 추가
Actor	Actor-based
Purpose	Constraint를 추가한다.
Overview	User가 Constraint 추가를 요청하면 Constraint를 추가한다.
Type	Primary
Cross Reference	Functions: R1.7 Use Cases:
Pre-Requisites	RV가 존재해야 한다.
Typical Courses of Events	(A):Actor, (S):System 1. (A):Constraint 추가 버튼(F)을 누른다. 2. (S):입력 받은 Category_Id와 일치하는 Category 객체를 찾는다. 3. (S):입력 받은 RV_Id와 일치하는 RV 객체를 찾는다. 4. (S):Input_Constraint의 값에 따라 RV객체에 Constraint를 추가한다.
Alternative Courses of Events	N/A
Exceptional Courses of Events	1. Input_Constraint가 Single 혹은 Error면, RV객체의 Single_Error를 변경한다. 2. Input_Constraint가 Property면, RV 객체의 Property_List에 Constraint_Name을 추가하고 Input_Manager.Property_List를 갱신한다.

Define Real Use Cases(7/11)

Use Case	Constraint 추가
Exceptional Courses of Events	3. Input_Constraint가 If이고, Input_Manager.Property_List에 Constraint_Name과 일치하는 Property가 존재하면, RV객체의 If_List에 Constraint_Name을 추가한다.

Define Real Use Cases(8/11)

Use Case	Constraint 삭제
Actor	Actor-based
Purpose	Constraint를 삭제한다.
Overview	User가 Constraint 삭제를 요청하면 Constraint를 삭제한다.
Type	Primary
Cross Reference	Functions: R1.8 Use Cases:
Pre-Requisites	Category 객체가 존재해야 한다.
Typical Courses of Events	(A) : Actor (S) : System 1. (A) : Constraint 삭제 버튼(I)을 누른다. 2. (S) : 입력받은 Category_Id와 일치하는 Category 객체를 찾는다. 3. (S) : 입력받은 RV_Id와 일치하는 RV 객체를 찾는다. 4. (S) : Select_Constraint의 값에따라, RV객체의 Constriant를 삭제한다.
Alternative Courses of Events	N/A

Define Real Use Cases(8/11)

Use Case	Constraint 삭제
Exceptional Courses of Events	<ol style="list-style-type: none">1. Select_Constraint가 Single 혹은 Error면, RV객체의 Single_Error를 변경한다.2. Select_Constraint가 Property면, Constraint_Name에 해당하는 모든 If_Constraint를 삭제하고 Input_Manager객체의 Property_List를 갱신하고, RV객체의 Property_List에서 Constraint_Name을 삭제한다..3. Input_Constraint가 If이면, RV객체의 If_List에서 Constraint_Name을 삭제한다.

Define Real Use Cases(9/11)

Use Case	대포갯 선택
Actor	Actor-based
Purpose	대포갯을 선택한다.
Overview	User가 대포갯을 선택, 해제하면 대포갯의 선택 여부를 수정한다.
Type	Primary
Cross Reference	Functions: R1.9 Use Cases:
Pre-Requisites	대포갯이 존재해야 한다.
Typical Courses of Events	(A) : Actor, (S) : System 1. (A) : RV 목록 중 하나를 선택 (D) 한다. 2. (S) : Input_Manager객체의 Selected_RV_List를 갱신한다.
Alternative Courses of Events	N/A
Exceptional Courses of Events	N/A

Define Real Use Cases(10/11)

Use Case	Test Case 계산
Actor	Actor-based
Purpose	Test Case를 계산한다.
Overview	User가 계산을 요청하면 Test Case를 계산한다.
Type	Primary
Cross Reference	Functions: R2.1 Use Cases:
Pre-Requisites	Category, 대푯값, Constraint가 입력되어야 한다.
Typical Courses of Events	(A) : Actor, (S) : System 1. (A) : Calculate 버튼(J)을 누른다. 2. (S) : Input_Manager의 Category_List와 Selected_RV_List를 이용하여 All_TC와 Selected_TC를 계산한다. 3. (S) : Selected_TC를 Display Field(M)에 출력한다. 4. (S) : All_TC를 File로 출력한다.
Alternative Courses of Events	N/A
Exceptional Courses of Events	N/A

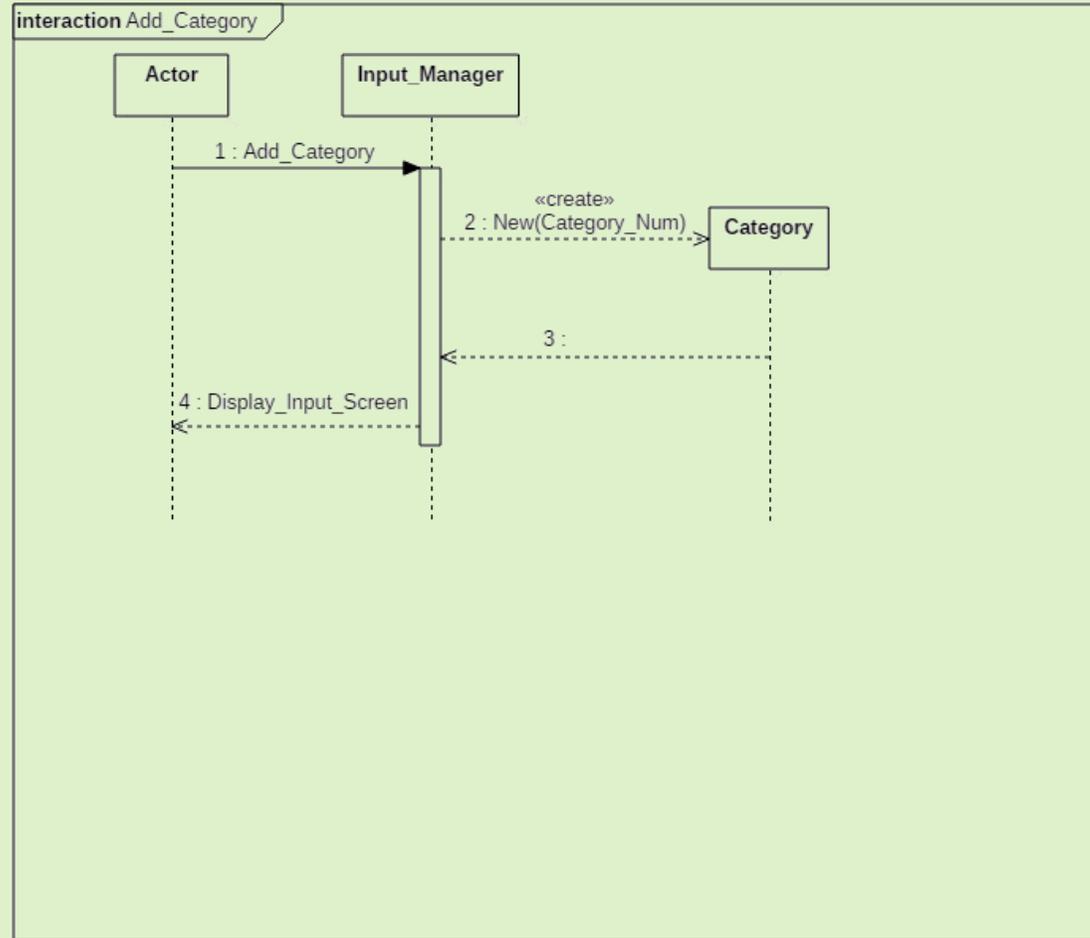
Define Real Use Cases(11/11)

Use Case	초기화
Actor	Actor-based
Purpose	모든 입력과 출력 화면을 초기화한다.
Overview	User가 초기화를 요청하면 입력과 출력화면을 초기화 한다..
Type	Primary
Cross Reference	Functions: R3.1 Use Cases:
Pre-Requisites	Category, 대푯값, Constraint가 입력되어야 한다.
Typical Courses of Events	(A) : Actor (S) : System 1. (A) : Clear 버튼(K)을 누른다. 2. (S) : Input_Manager에서 모든 입력 내용을 삭제하고 입력 창을 비워준다. 3. (S) : Output_Manager에서 출력 창을 비워준다.
Alternative Courses of Events	N/A
Exceptional Courses of Events	N/A

Define Interaction Diagrams

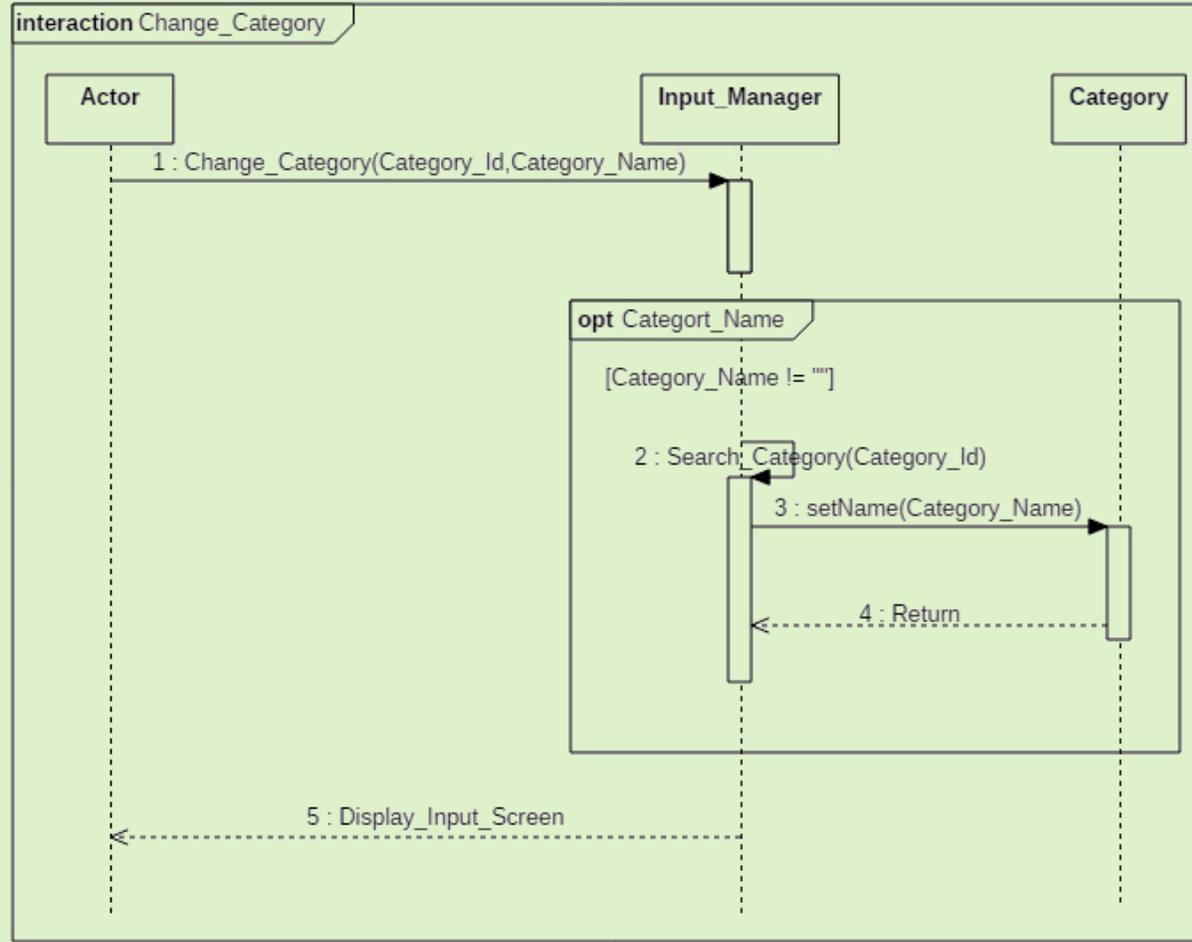
Category 추가(1/11)

Define Interaction Diagrams



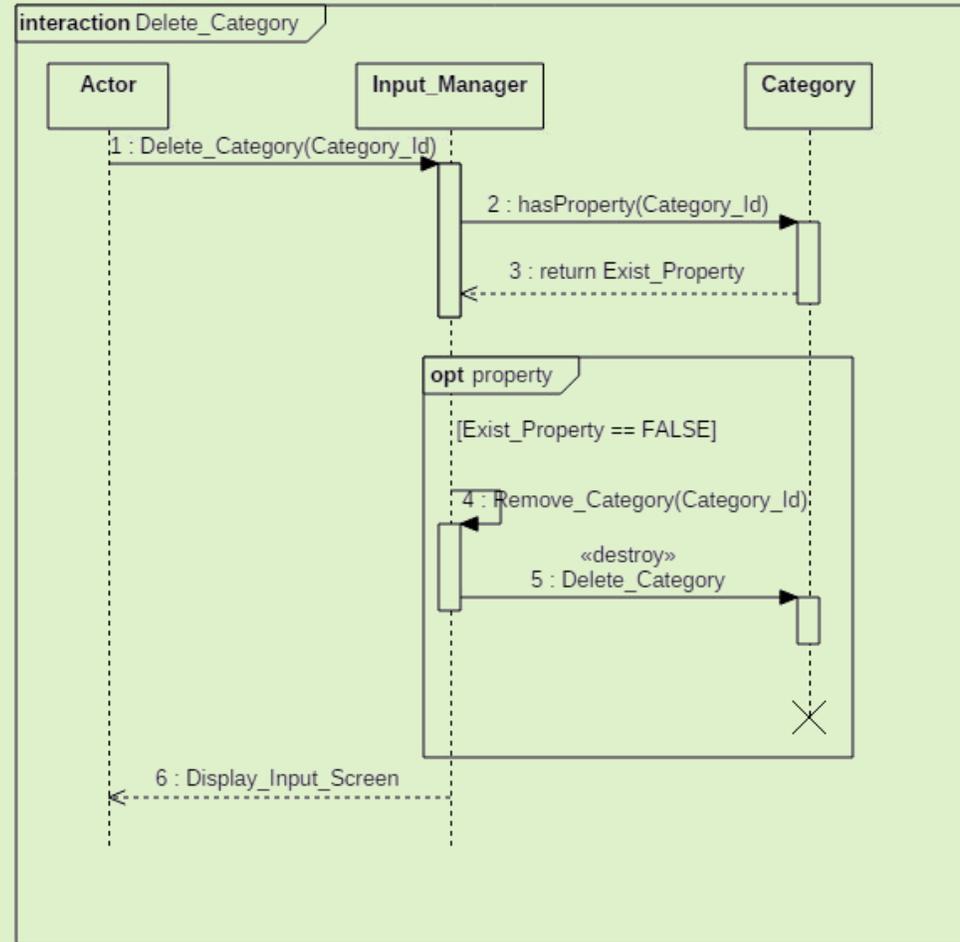
Category 수정(2/11)

Define Interaction Diagrams



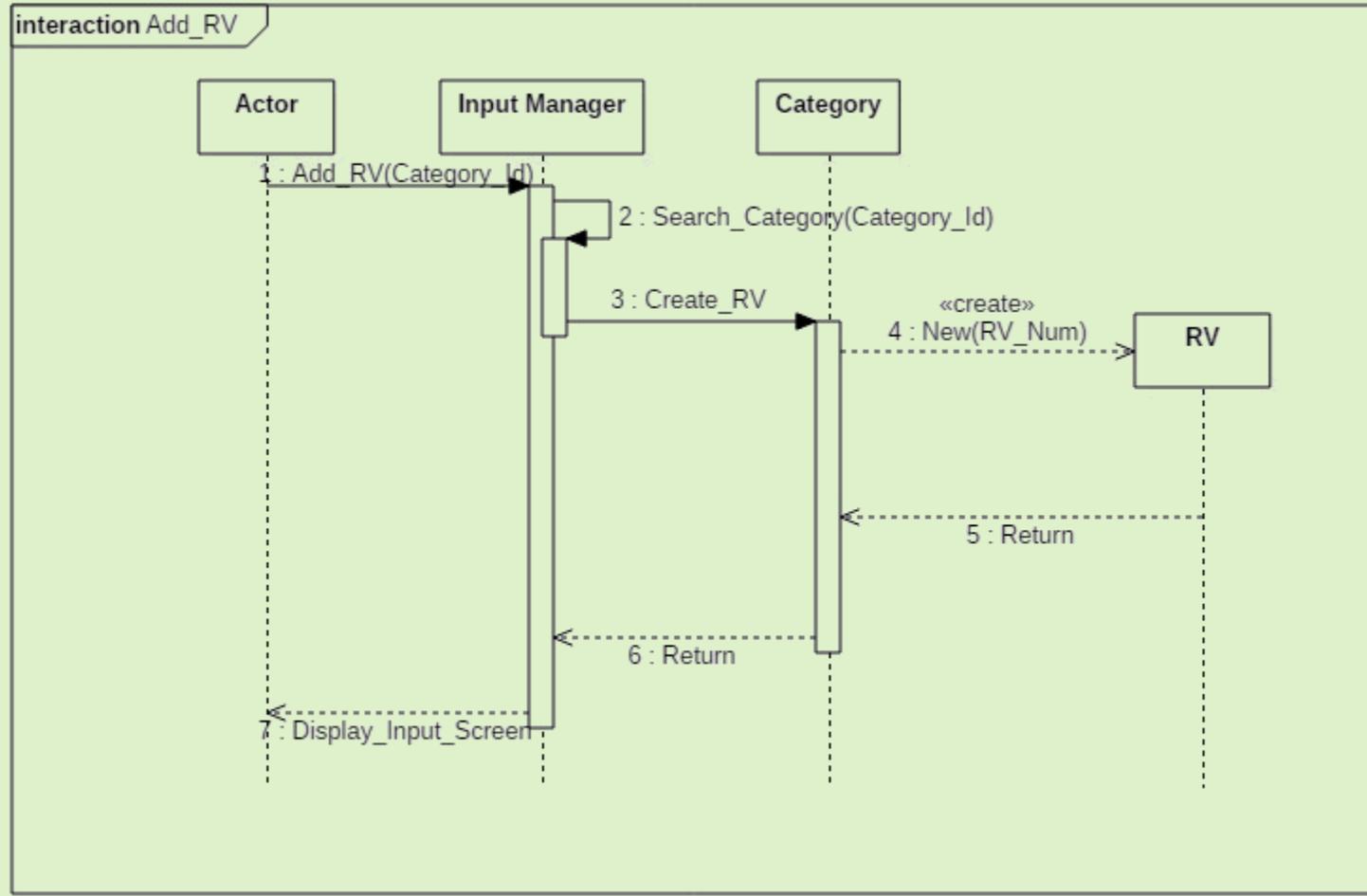
Category 삭제(3/11)

Define Interaction Diagrams



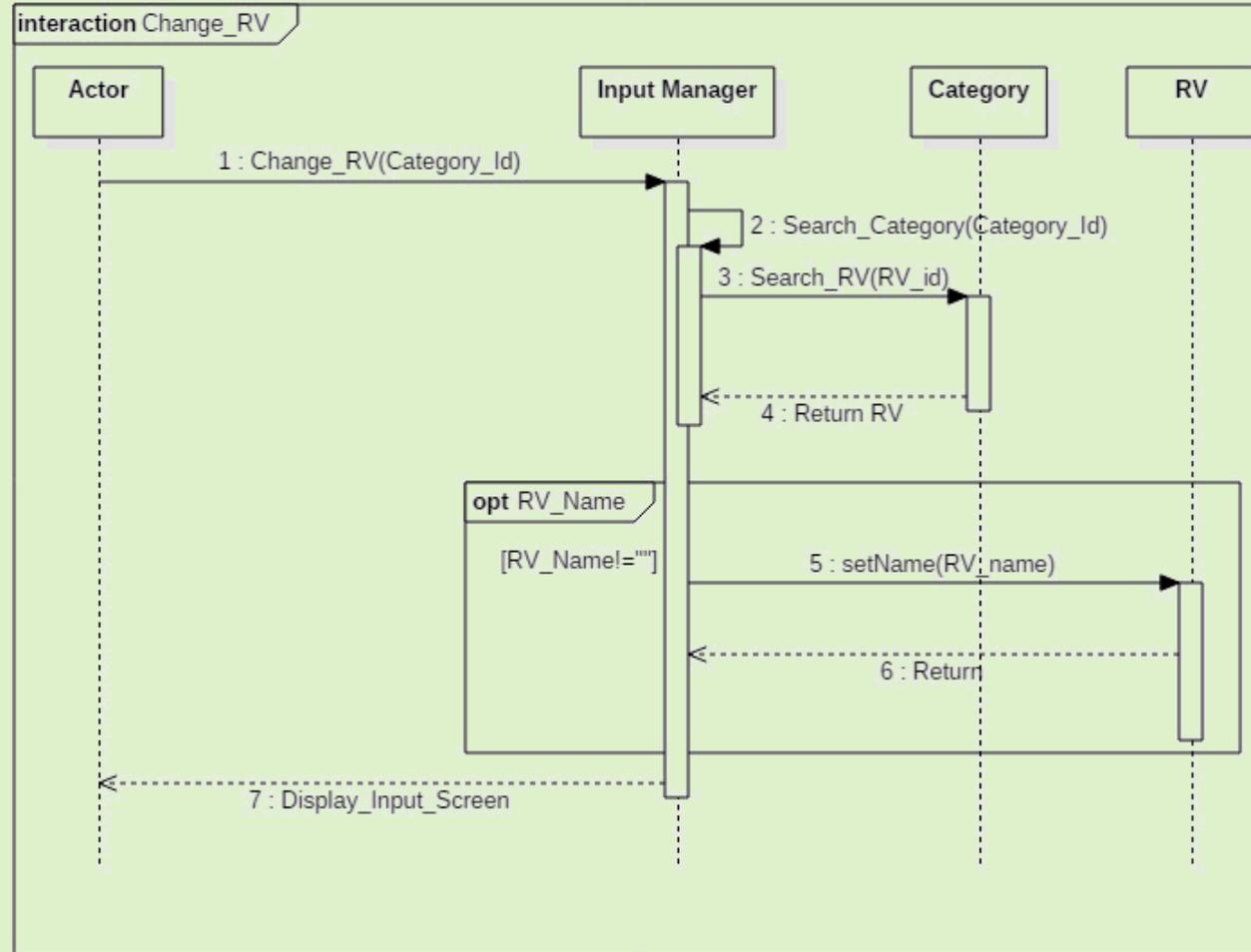
대포켓값 추가(4/11)

Define Interaction Diagrams



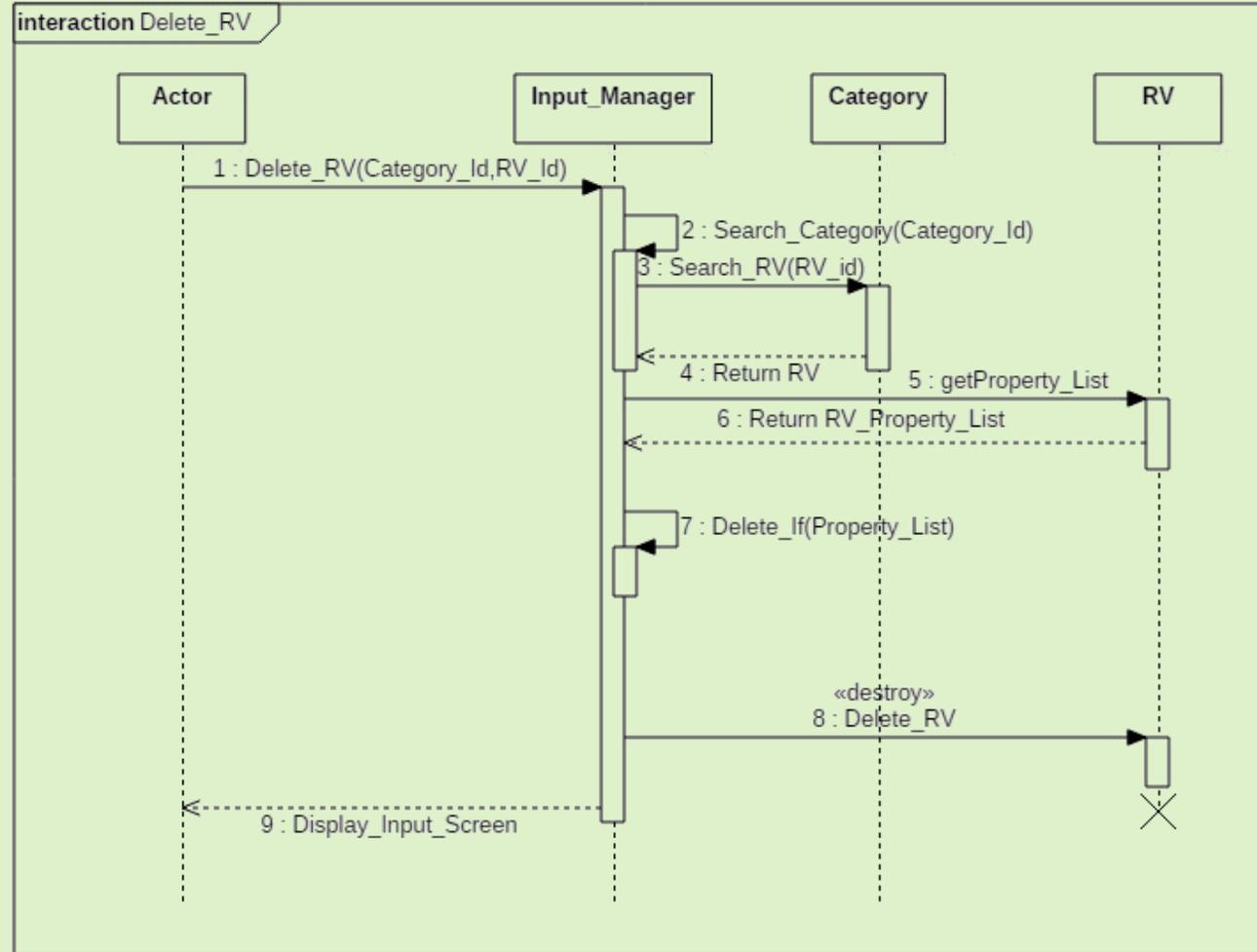
대포깁 수정(5/11)

Define Interaction Diagrams



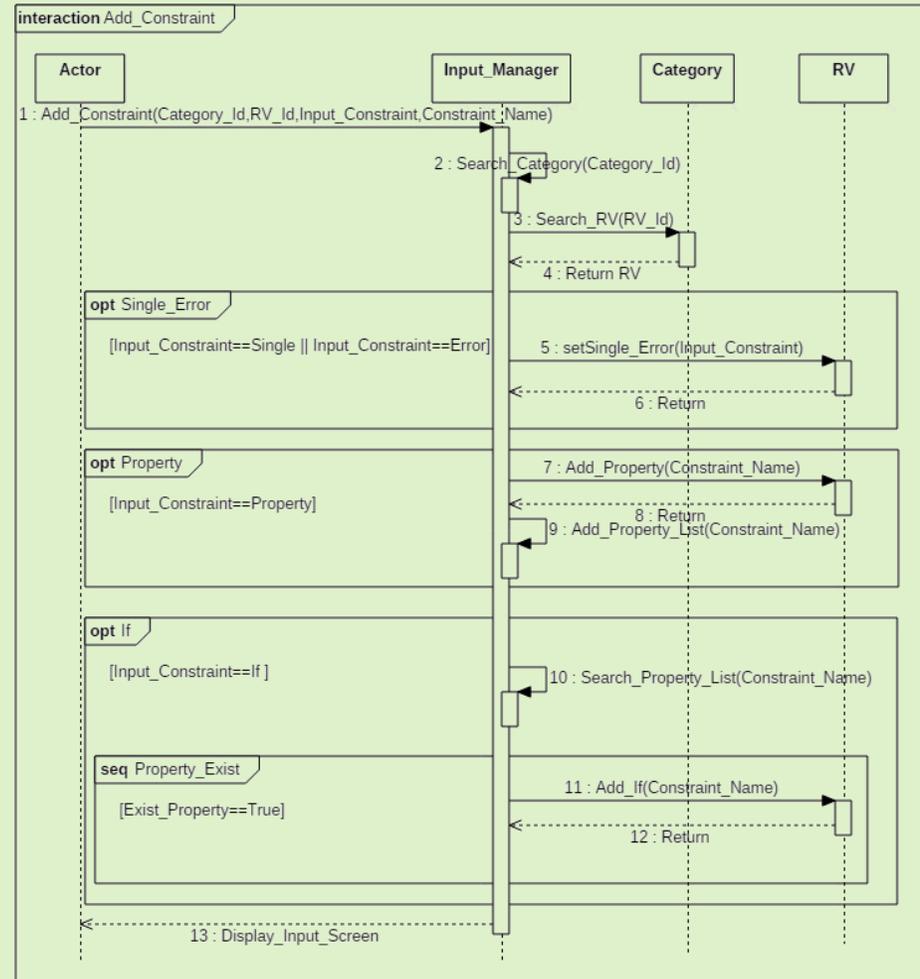
대포깁 삭제(6/11)

Define Interaction Diagrams



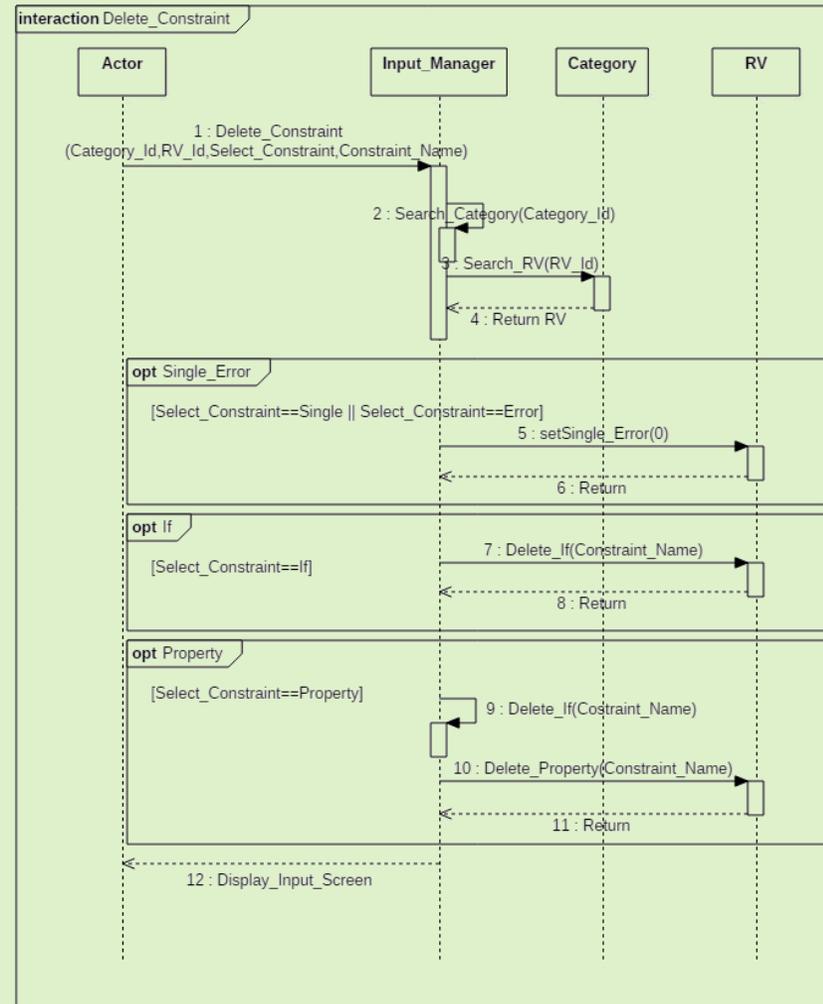
Constraint 추가(7/11)

Define Interaction Diagrams



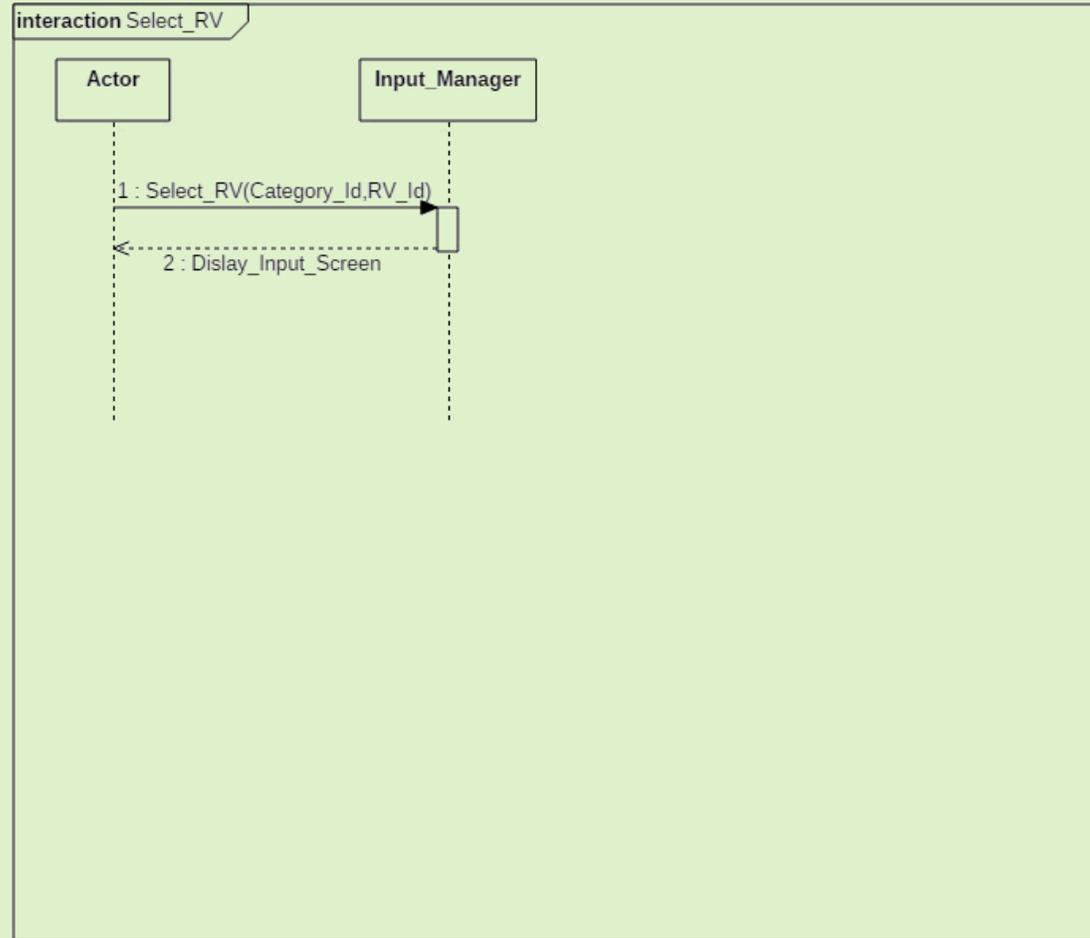
Constraint 삭제(8/11)

Define Interaction Diagrams



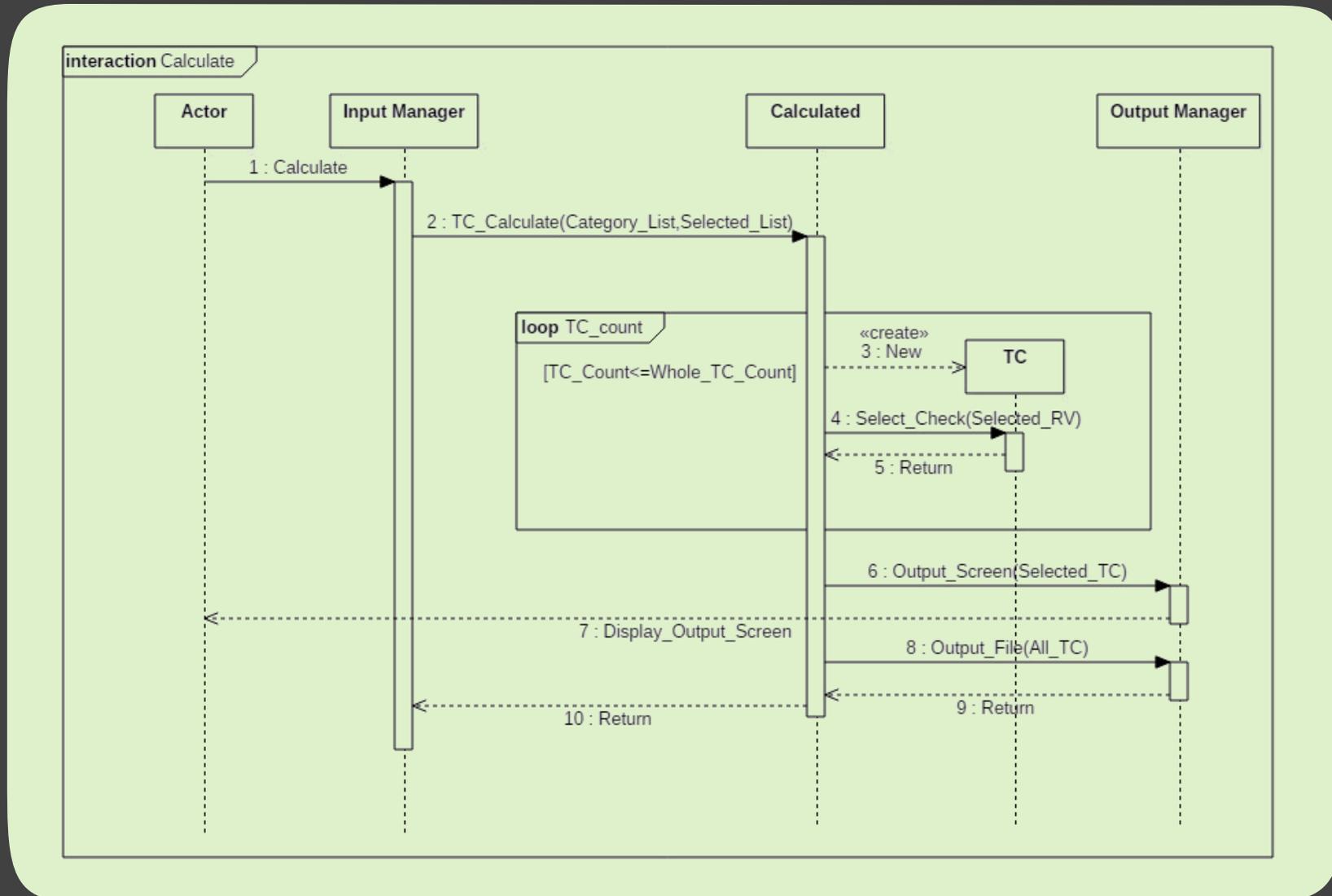
대포켓값 선택(9/11)

Define Interaction Diagrams



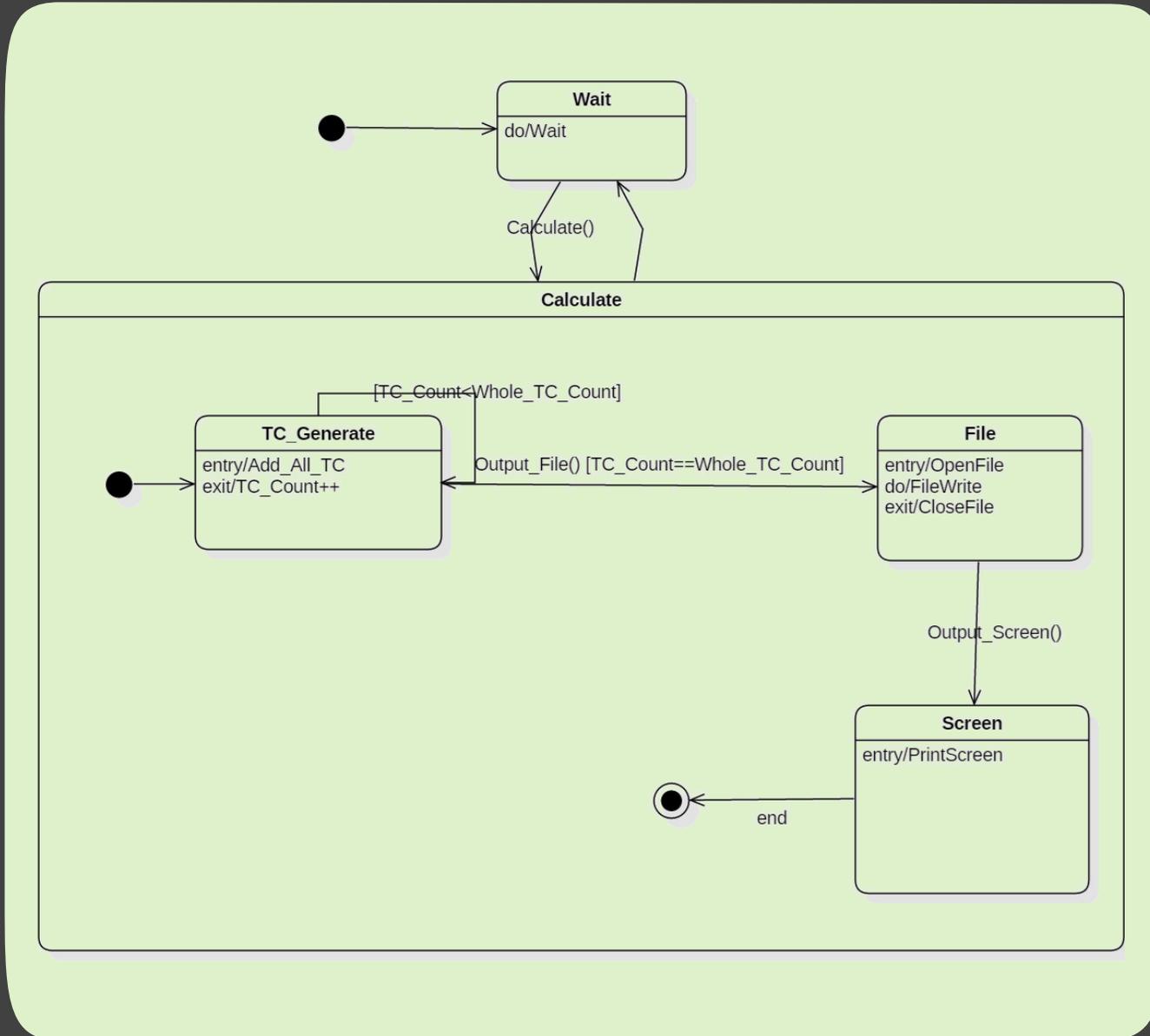
Test Case 계산(10/11)

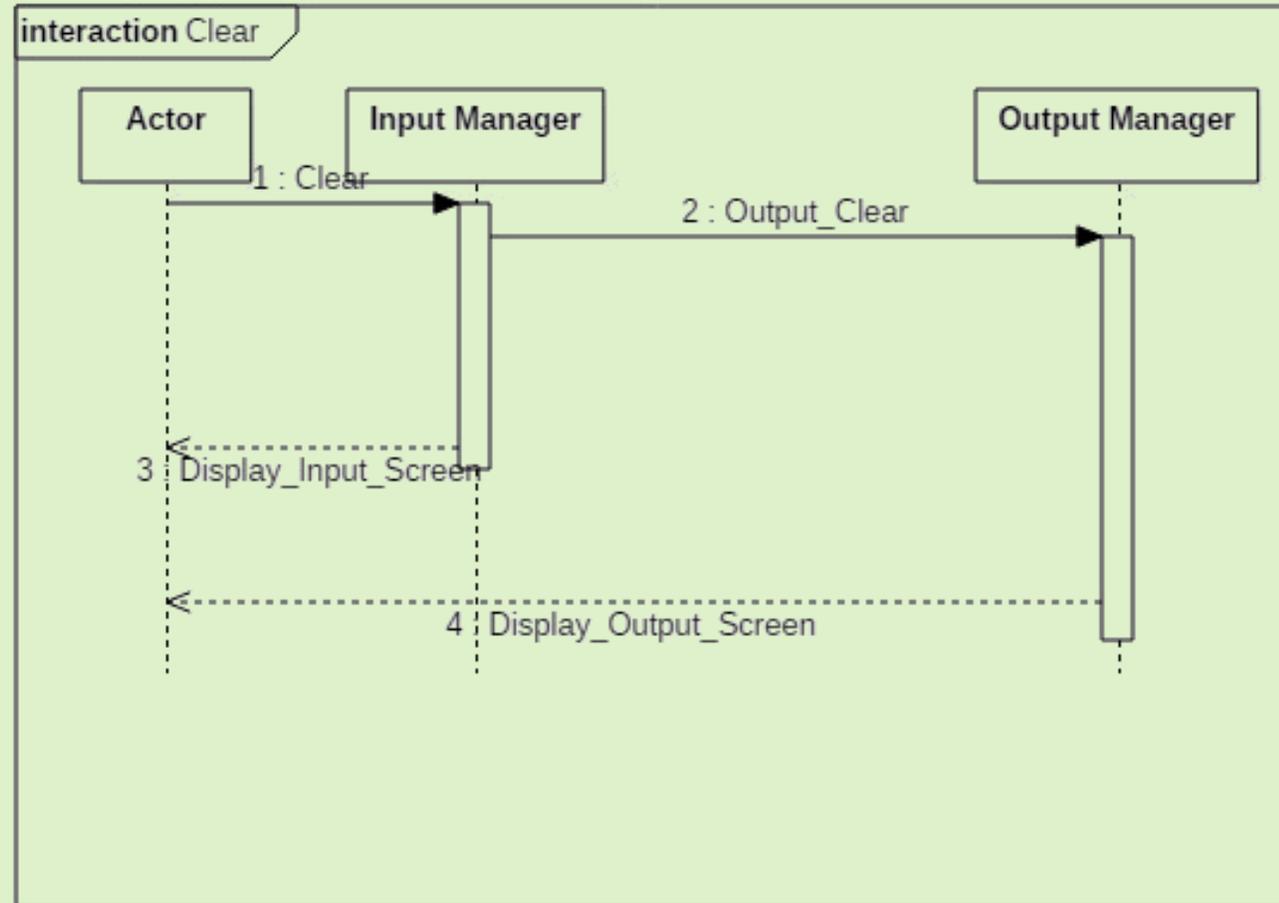
Define Interaction Diagrams



Test Case 계산(10/11)

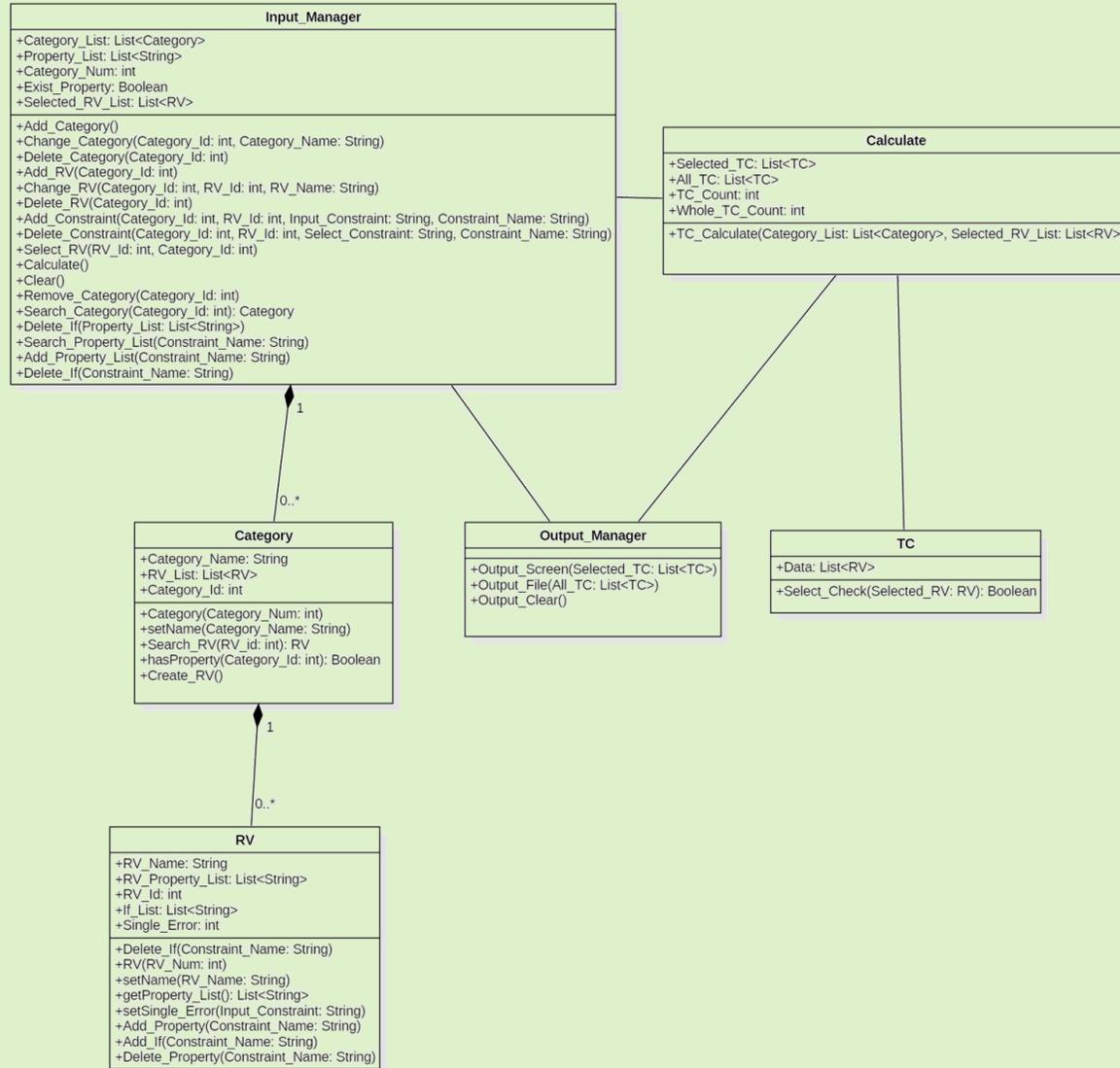
State Diagram





Define Design Class Diagrams

Define Design Class Diagrams



Design Traceability Analysis

Design Traceability Analysis

Operation in sequence diagram	operation in interaction diagram	method	Class
1. Category 추가	Add_Category	Add_Category():void	Input_Manager
2. Category 수정	Chage_Category	Change_Category(Category_Id:int, Category_Name:String):void	
3. Category 삭제	Delete_Category	Delete_Catoegry(Category_Id:int):void	
4. 대푯값 추가	Add_RV	Add_RV(Category_Id:int):void	
5. 대푯값 수정	Change_RV	Change_RV(Category_Id:int,RV_Id:int,RV_Name:String):void	
6. 대푯값 삭제	Delete_RV	Delete_RV(Category_Id:int,RV_Id:int):void	
7. Constraint 추가	Add_Constraint	Add_Constraint(Category_Id:int,RV_Id:int,Input_Constraint:String,Constraint_Name:String):void	
8. Constraint 삭제	Delete_Constraint	Delete_Constraint(Category_Id:int,RV_Id:int,Select_Constraint:String,Constraint_Name:String):void	
9. 대푯값 선택	Select_RV	Select_RV(Category_Id:int,RV_Id:int):void	
10. Test Case 계산	Calculate	Cacluate():void	
11. 초기화	Clear	Clear():void	
		Remove_Category(Category_Id:int):void	Category
		Search_Category(Category_Id:int):Category	
		Delete_If(Property_List:List<String>):void	
		Search_Property_List(Constraint_Name:String):void	
		Add_Property_List(Constraint_Name:String):void	RV
		Delete_If(Constraint_Name:String):void	
		setName(Category_Name:String):void	
		Search_RV(RV_Id:int):RV	
		hasProperty(Category_Id:int):Boolean	
		Create_RV():void	
		Delete_if(Constraint_Name:String)	
		setName(RV_name:String):void	
		getProperty_List():List<String>	
		setSingle_Error(Input_Constraint:String):void	
		Add_Property(Constraint_Name:String):void	Output_Manager
		Add_If(Constraint_Name:String):void	
		Delete_Property(Constraint_Name:String):void	
		Output_Screen(Selected_TC:List<TC>):void	
		Output_File(ALL_TC:List<TC>):void	Calculate
		Output_Clear():void	
		TC_Calculate(Category_List:List<Category>,Selected_List:List<Category>):void	TC
		Select_Check(Selected_RV:List<RV>):Boolean	

Q & A

Thank you ~!