

# Category Partition Testing Tool

201414134 오세욱

201414136 임현유

201211375 임동현

201211387 하현규

# Index

---

**1. Project Introduction**

**2. Requirements**

**3. Use Cases**

**4. Detailed Plan**

**5. Q&A**

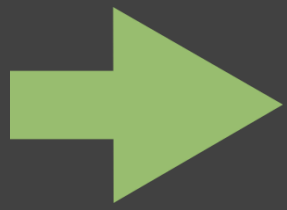
# Project Introduction

Motivation

Alternative Solution

Prototype

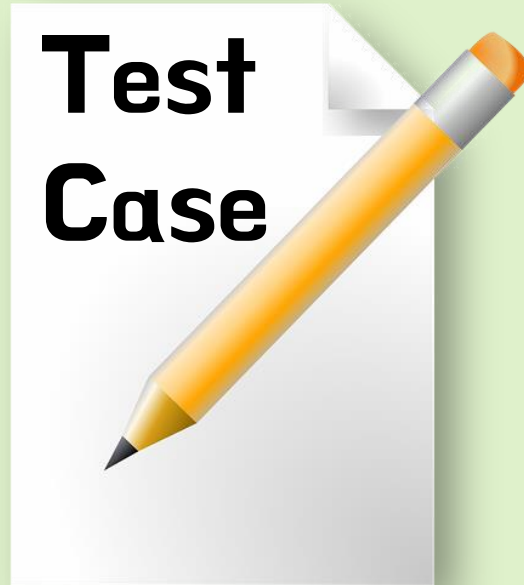
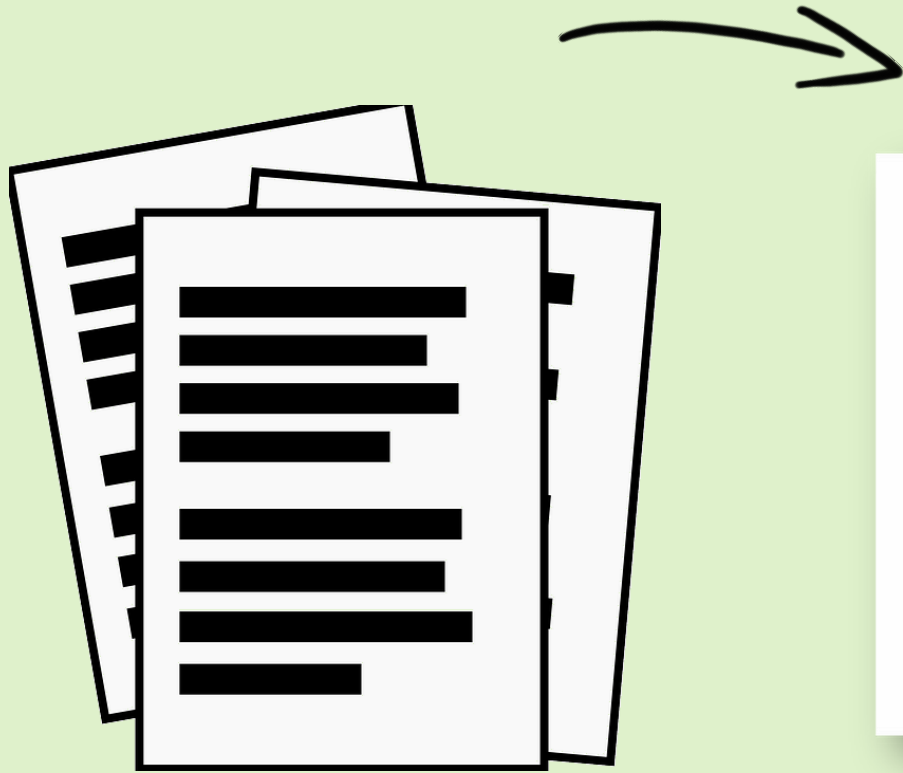
000



유망~!

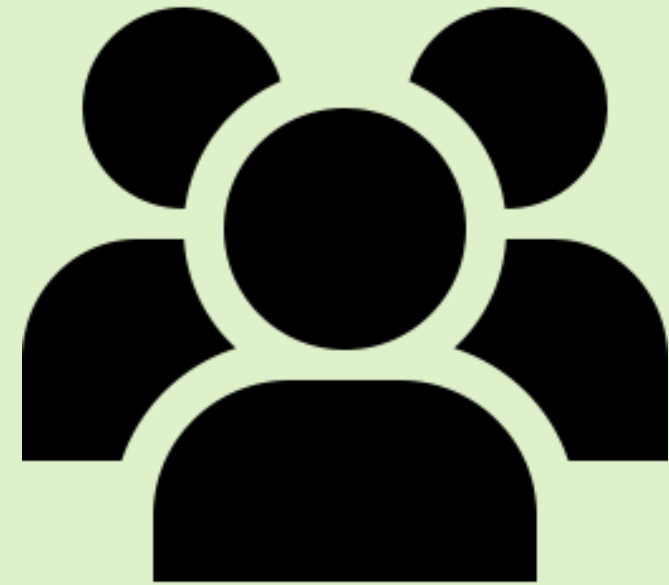
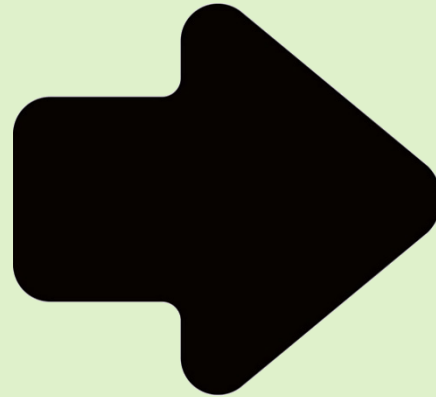
# Motivation

# Project Introduction



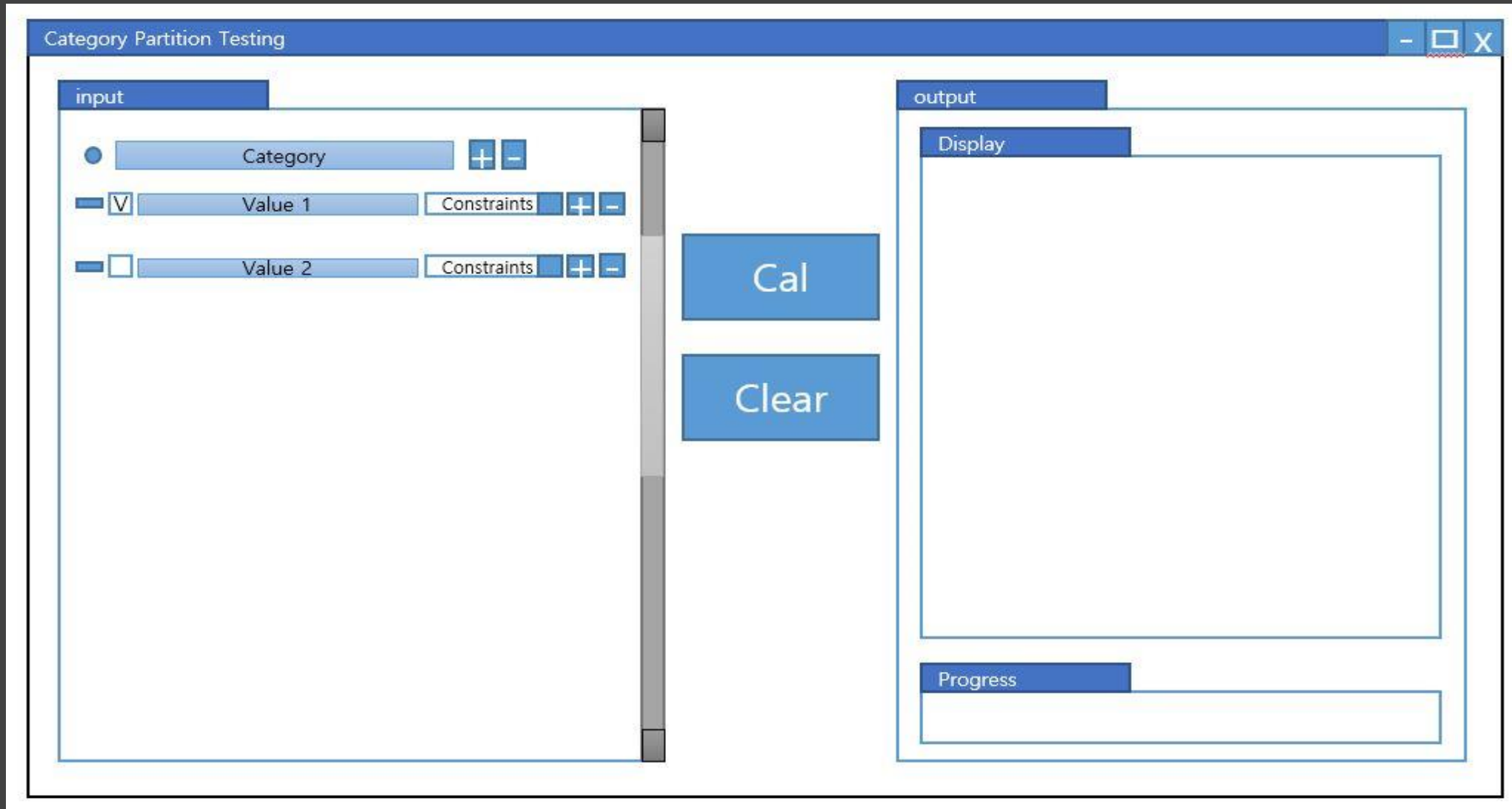
# Alternative Solution

# Project Introduction



# Prototype

# Project Introduction



# Requirements

Functional Requirements

Non-Functional Requirements



# Functional Requirement

## Requirements

Function	Description
Category 추가	Category 객체를 생성한다.
Category 삭제	Category 객체를 삭제한다.
Category 수정	Category 객체를 수정한다.
Value 추가	Value 객체를 생성한다.
Value 삭제	Value 객체를 삭제한다.
Value 수정	Value 객체를 수정한다.
Constraints 추가	Constraints를 추가한다.
Constraints 삭제	Constraints를 삭제한다.
Constraints 수정	Constraints를 수정한다.
Value 우선 순위 선택	Value에 우선 순위를 부여한다.

# Functional Requirement

## Requirements

Function	Description
Test Case 계산	전체 Test Case를 계산한다.
우선 순위 계산	우선 순위가 있는 Value에 관련된 Test Case를 먼저 계산하고 그 Test Case가 전체 Test Case 중 가지는 비중을 계산한다.
진행 상황 계산	전체 Test Case중 얼마큼 진행 했는지 '%'로 출력한다.
출력	결과값들을 출력한다.
우선 순위 출력	우선 순위 Test Case를 화면에 출력한다.
진행 상황 출력	진행 상황을 화면에 출력한다.
File 출력	전체 Test Case를 File로 출력한다.
초기화	모든 입력과 출력 화면을 초기화한다.
프로그램 종료	프로그램을 종료한다.

**누구든 조작하기 쉽게**

**정확한 TEST CASE를**

**신속하게 뽑아 보기 좋게 !!**

# System Architecture

System Architecture

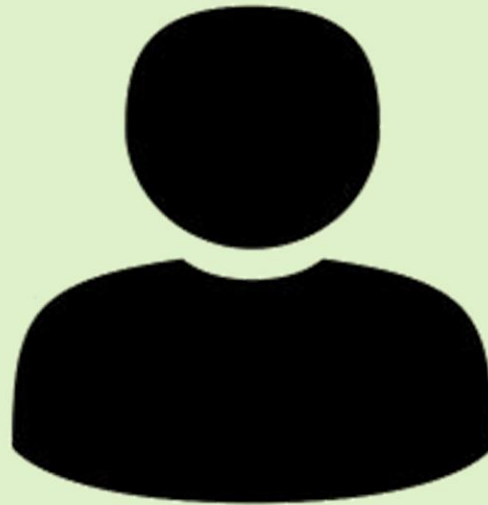
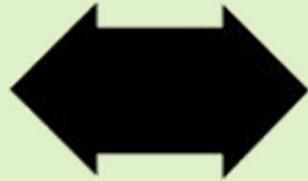
Use case Diagram

# System Architecture

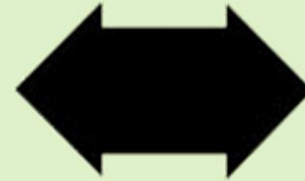
# System Architecture



Programmer



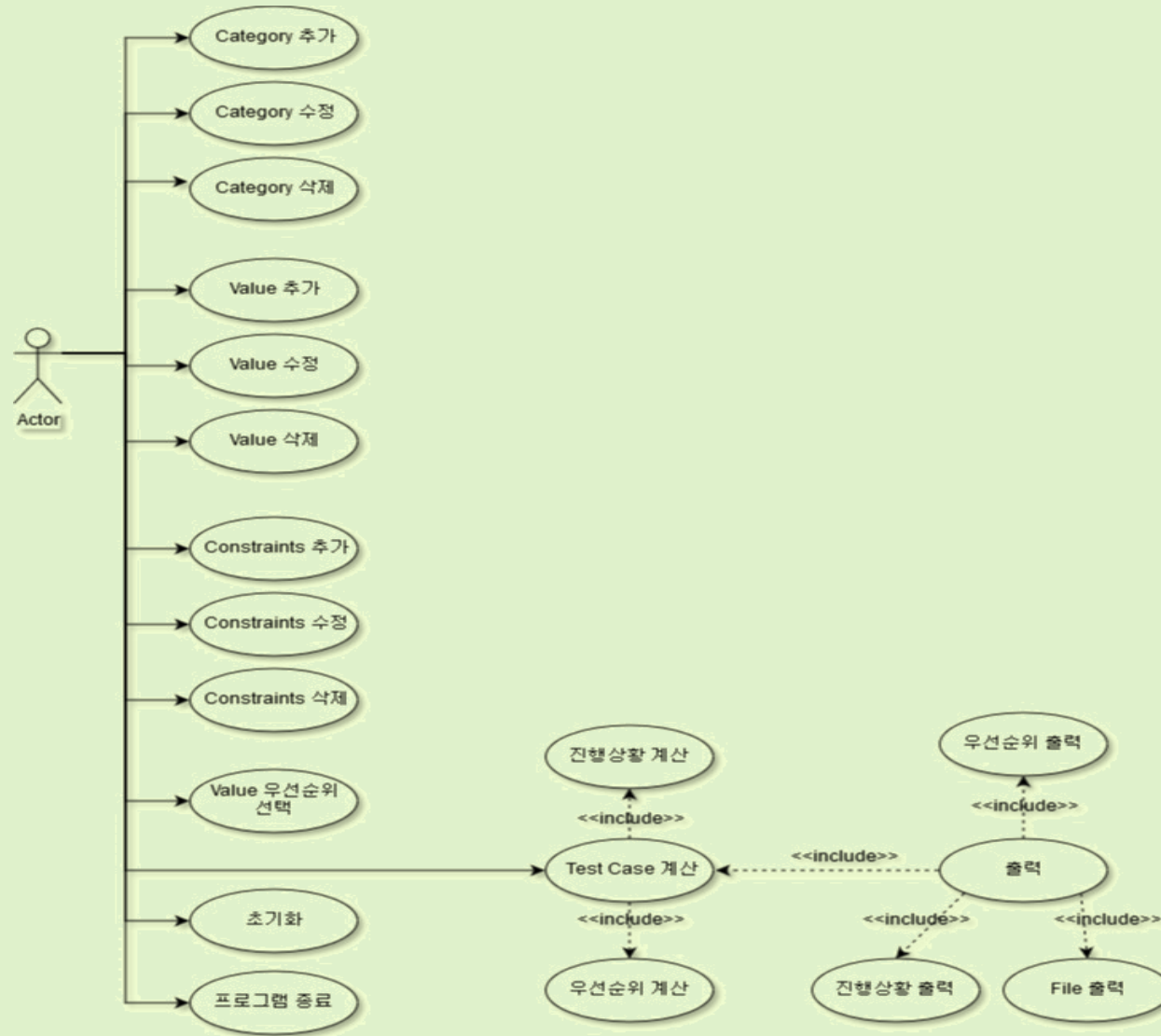
Tester



System

# Use Case Diagram

# System Architecture



# Detailed Plan

Risk Analysis  
Schedule

# Risk Analysis

## Detailed Plan

Risk	Probability(5)	Significance	Weight
Java 숙련도 부족	4	5	20
UML 숙련도 부족	5	3	15
고양이가 아픴	2	4	8
시험 기간	5	5	25
아르바이트	2	3	6



Risk	Way of Reduction
Java 숙련도 부족	이전에 사용한 자바 교재와 Google을 활용한다.
UML 숙련도 부족	UML Manual과 수업 ppt를 활용한다.
고양이가 아픴	병원을 꾸준히 데려가고, 평소에 잘해준다.
시험 기간	평소에 미리 공부한다.
아르바이트	그만 둔다.



# Schedule

# Detailed Plan

Stage	Phase(00x0)/Activity(000x)	Schedule(Week)									
		1	2	3	4	5	6	7	8	9	10
1000. Plan & Elabo rate	1001. Define Draft Plan	█									
	1002. Create Preliminary Investigation Report										
	1003. Define Requirements	█									
	1004. Record Terms in Glossary	█	█								
	1005. Implement Prototype		█								
	1006. Define System Architecture		█								
	1007. Define Use Cases		█								
	1008. Define Draft Conceptual Model		█								
	1009. Define Draft System Architecture		█								
	1010. Refine Plan		█								
2000. Build	2010. Revise Plan		█	█							
	2020. Synchronize Artifacts			█							
	2030. Analyze										
	2031. Define Essential Use Case			█							
	2032. Refine Use Case Diagrams			█							
	2033. Refine Conceptual Model			█							
	2034. Refine Glossary			█							
	2035. Define System Sequence Diagrams			█	█						
	2036. Define Operation Contracts			█							
	2037. Define State Diagrams			█							
	2040. Design										
	2041. Define Real Use Cases				█						
	2042. Define Reports, UI and Storyboards				█	█					
	2043. Refine System Architecture				█						
	2044. Define Interaction Diagrams				█	█					
	2045. Define Design Class Diagrams				█	█	█				
	2046. Define Database Schema				█						
	2050. Construct										
	2051. Implement Class & Interface Definition							█			
	2052. Implement Methods							█			
	2053. Implement Windows							█			
	2054. Implement Reports							█			
	2055. Implement DB Schema							█			
	2056. Write Test Code							█			
	2060. Test										
	2061. Unit Testing								█		
2062. Integration Testing								█			
2063. System Testing								█			
2064. Performance Testing								█			
2065. Acceptance Testing								█			
2066. Documentation Testing								█			

**Q & A**

Thank you ~!