

OOPT Stage 1000

<Plan and Elaboration>

Team. T2

201411278 서희진

201411304 이지수

201411317 조민규

201213189 박성규

Stage 1000. Plan and Elaboration

Activity 1001. Define Draft Plan	3
Activity 1002. Create Preliminary Investigation Report.....	5
Activity 1003. Define Requirements.....	7
Activity 1004. Record Terms in Glossary	10
Activity 1005. Implement Prototype	11
Activity 1006. Define Draft System Architecture.....	13
Activity 1007. Define Business Use Case.....	14
Activity 1008. Define Business Concept Model	21
Activity 1009. Define System Test Case	22
Activity 1010. Refine Plan	23

Activity 1001. Define Draft Plan

1. Motivation

소프트웨어의 기획개발 업무는 주로 팀으로 구성되어 프로젝트를 진행하는 경우가 많고, 그렇기에 팀 간 문서 공유는 필수적이다. 또한, 소프트웨어의 블랙 박스 테스트를 위해 Category 를 만들고 직접 조합하여 Test Case 를 작성하는 과정에서, 보통 숫자로 넘버링하여 조합하기에 Test Case 의 가독성이 좋지 않다. 이러한 문제를 해결하기 위해 구글드라이브를 활용하여 빠른 문서공유가 가능하고, 깔끔한 UI 를 활용하여 조합된 Test Case 를 한눈에 알아볼 수 있는 Category Partitioning Testing Tool 이 필요하다고 판단하였다.

2. Project Objectives

- 수동으로 Test Scripts 를 만드는 것은 많은 시간이 걸리고 비효율적인 일이기 때문에 Test Specification 을 기준으로 Test Scripts 를 만들어주는 Category Partitioning Testing Tool 을 개발한다.
- 프로젝트를 팀으로 진행하는 경우가 많기 때문에 팀 간의 빠른 문서 공유를 위해 구글드라이브와 연동한다.

3. Functional Requirements

- Load
- LoadFile
- LoadDrive
- AnalyzeFile
- AnalyzeCategory
- AnalyzeValue
- AnalyzeConstraints
- AnalyzeComment
- MakeCase
- Display
- Save
- SaveFile
- SaveDrive
- Exit

4. Non-Function Requirements

- A. 직관적인 인터페이스
- B. 텍스트 파일로 작성된 Test Specification 만 분석한다.

5. Resource Estimation

- A. Human Resource : 4 명
- B. Project Duration : 3 개월(12 주)
- C. Human Efforts(Man-Month) : 12
- D. Cost : 식대 1,200,000 원 (5,000 원/일 × 4 명 × 5 일/주 × 12 주)

6. Other Information

- A. Future Version

Category Partitioning Testing Tool 을 이용하여 나오는 Output 을 구글드라이브 뿐만 아니라 블로그나 카페와 같은 다른 매체에도 자동으로 업로드 하여, 프로젝트를 진행하는 팀원들이 시간이나 장소 뿐만 아니라 매체의 제약 없이 편리하게 접근 할 수 있도록 확장한다.

Activity 1002. Create Preliminary Investigation Report

1. Alternative Solutions

- A. 개발 전문 업체에 프로그램의 제작을 의뢰한다.
- B. 주변 인맥을 활용하여 해결한다.
- C. Category Partitioning Tool 에 대한 오픈소스를 활용한다.

2. Project Justification(Business Demands)

- A. Cost : 식대 외의 다른 비용이 필요하지 않아 다른 대안보다 훨씬 저렴하다.
- B. Duration : 3 개월(12 주)
- C. Risk : 개발 능력의 부족, 팀원의 이탈, 디자인 능력의 부족, 팀원의 고양이의 지나친 애교, 지나친 굶주림 및 과제의 중첩, 과도한 음주 및 숙취
- D. Effect
 - 구글 드라이브에서 파일을 불러와서 CPT 을 수행 한 후 결과물을 출력 및 업로드 할 수 있다.
 - 기존의 프로그램을 결과물을 따로 복사해 저장하는 과정이 필요했다면, 해당 프로그램은 자동으로 결과물을 문서화 시켜 시간을 절약할 수 있다.

3. Risk Management

Risk	Probability	Significance	Weight
개발 능력의 부족	5	10	50
팀원의 이탈	2	5	10
디자인 능력의 부족	2	2	4
고양이의 지나친 애교	5	1	5
지나친 굶주림	4	4	16
과제의 중첩	5	5	25
과도한 음주 및 숙취	4	3	12

4. Risk Reduction Plan

Risk	Way of Reduction
개발 능력의 부족	주변 동료의 소스코드를 활용하거나, 적극적인 수업 참여와 관련 도서 및 인터넷을 통하여 학습한다.
팀원의 이탈	잘 다독이고 맛있는 걸 사먹여서 기분을 좋게 만든다.
디자인 능력의 부족	많은 참고도서를 참고하여 디자인 한다.
고양이의 지나친 애교	고양이의 까까를 물려주고 나오거나, 팀원의 집에서 프로젝트를 진행한다.
지나친 굶주림	매 만남마다 점심이나 저녁을 꼬박꼬박 챙겨 먹는다.
과제의 중첩	주변 동료의 과제를 활용하여 효율적으로 과제를 마무리한다.
과도한 음주 및 숙취	음주 전 여명이나 컨디션을 음용한다.

5. Market Analysis

무료로 제공받을 수 있는 CPT 도구가 거의 없고, 단편적인 기능만을 가진 CPT 도구가 많아서 시장성이 높다.

6. Other Managerial Issues

- A. 2017 년 6 월 안에 프로젝트가 종료되어야 한다.
- B. 추후에 졸업작품으로 이용할 수 있다.

Activity 1003. Define Requirements

1. Functional Requirements

Function	Description
Load	파일을 불러온다
LoadFile	컴퓨터에 존재하는 txt 파일을 불러온다.
LoadDrive	구글드라이브에 존재하는 txt 파일을 불러온다.
AnalyzeFile	불러온 파일을 분석한다.
AnalyzeCtg	불러온 파일의 Category 를 분석한다.
AnalyzeVal	불러온 파일의 value 를 분석한다.
AnalyzeCons	불러온 파일의 Constraints 를 분석한다.
AnalyzeComment	불러온 파일의 Comment 를 분석한다.
MakeCase	각 Constraints 에 따른 Test Case 를 조합한다.
Display	Category 와 Value 와 Constraints 와 Script 목록을 출력한다.
Save	Test Script 를 저장한다.
SaveFile	Test Script 를 컴퓨터에 저장한다.
SaveDrive	Test Script 를 구글드라이브에 저장한다.
Exit	프로그램을 종료한다.

Ref	Function	Category
R.1	Load	Evident
R.1.1	LoadFile	Evident
R.1.2	LoadDrive	Evident
R.2	AnalyzeFile	Hidden
R.2.1	AnalyzeCtg	Hidden
R.2.2	AnalyzeVal	Hidden
R.2.3	AnalyzeCons	Hidden
R.2.4	AnalyzeComment	Hidden
R.3	MakeCase	Hidden
R.4	Display	Evident
R.5	Save	Evident
R.5.1	SaveFile	Evident
R.5.2	SaveDrive	Evident
R.6	Exit	Evident

2. Performance Requirements

- A. Test Case 분석 작업은 5 초 이내로 수행되어야 한다.
- B. 구글드라이브에 업로드 하는 작업은 5 초 이내로 수행되어야 한다.

3. Operating Environments

Microsoft Windows 7 이상

4. Develop Environments

- A. 운영체제 : Windows 7, 10
- B. CPU : Intel
- C. IDE : Eclipse
- D. 개발 언어 : JAVA
- E. UML 툴 : StarUML

5. Interface Requirements

A. Main 화면

I. 파일 불러오기 버튼

클릭 시 컴퓨터 내의 파일을 불러오는 작업을 시작한다.

II. 구글드라이브 불러오기 버튼

클릭 시 구글드라이브 내의 파일을 불러오는 작업을 시작한다.

III. 종료하기 버튼

클릭 시 프로그램을 종료한다.

B. Category 화면

I. 카테고리 리스트

화면에 Category 들을 보여준다. Category 위에 마우스를 올리면 색을 바꾸고, 클릭 시에는 Value 들을 표현하는 화면으로 넘어간다.

II. 조합하기 버튼

분석된 내용을 이용하여 Test Case 를 조합한다.

III. 로고 버튼

메인 화면으로 돌아간다.

C. Value 화면

I. Value 리스트

화면에 Value 들을 보여준다. Value 위에 마우스를 올리면 해당 Value 의 색을 바꾸고 Constraints 들을 보여준다.

II. 로고 버튼

메인 화면으로 돌아간다.

III. 돌아가기 버튼

클릭 시 Category 화면으로 이동한다.

D. Test Script 화면

I. Test Case 리스트

화면에 Test Case 들을 보여준다. Test Case 위에 마우스를 올리면 해당 Test Case 의 색을 바꾸고, 세부 사항들을 보여준다.

II. 로고 버튼

메인 화면으로 돌아간다.

III. 저장하기 버튼

클릭 시 Test Script 를 저장하는 화면으로 이동한다.

E. 파일 저장 화면

I. 파일로 저장하기 버튼

버튼을 누르면 파일 경로를 선택하는 창이 뜨고, 원하는 파일명으로 저장된다.

II. 클라우드로 저장하기 버튼

버튼을 누르면 구글 로그인 창이 뜨고, 원하는 파일명으로 저장 및 업로드 된다.

6. Other Requirements

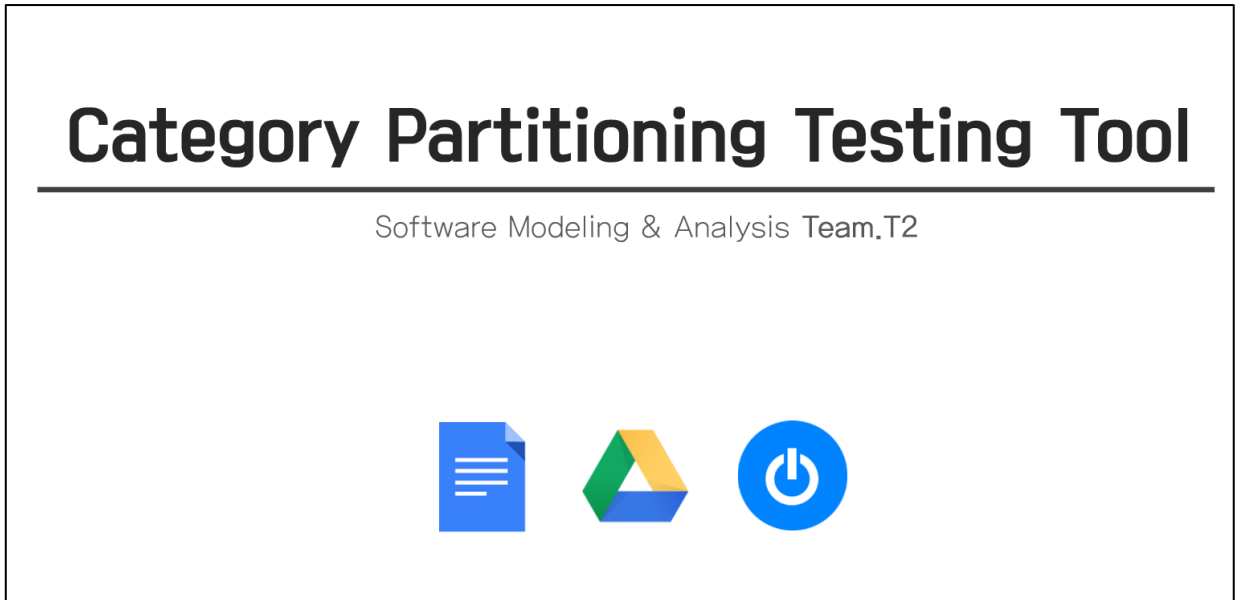
A. Google 계정을 필요로 한다.

Activity 1004. Record Terms in Glossary

Term	Description	Remark
Ctg	Category	
Val	특정 Category 에 해당하는 Value	
Cons	특정 Value 에 해당하는 Constraint	

Activity 1005. Implement Prototype

1. 메인 화면



2. 카테고리 출력 화면

Category Partitioning Testing Tool				30 Categories
Software Modeling & Analysis Team.T2				
Category				
이지수	미인규	SungKyu	Software	더 이상
조민규	서영규	Minkyu	Modeling	쓸
서희진	희이진	Jisoo	Analysis	말이
박성규	고양이	일식	Team	없어
지이수	Heejin	양식	2%	웃!

3. Value 화면 출력

Category Partitioning Testing Tool
Software Modeling & Analysis Team,T2

14 Choices

Category : 고양이

애옹애옹	Cute	1
왜옹왜옹	Noisy	2
야옹야옹	Happy	3
그릉그릉	Annoying	4

[If 만규바보] [Single]

돌아가기

4. Test Script 출력 화면

Category Partitioning Testing Tool
Software Modeling & Analysis Team,T2

270 Test Cases

Test Script

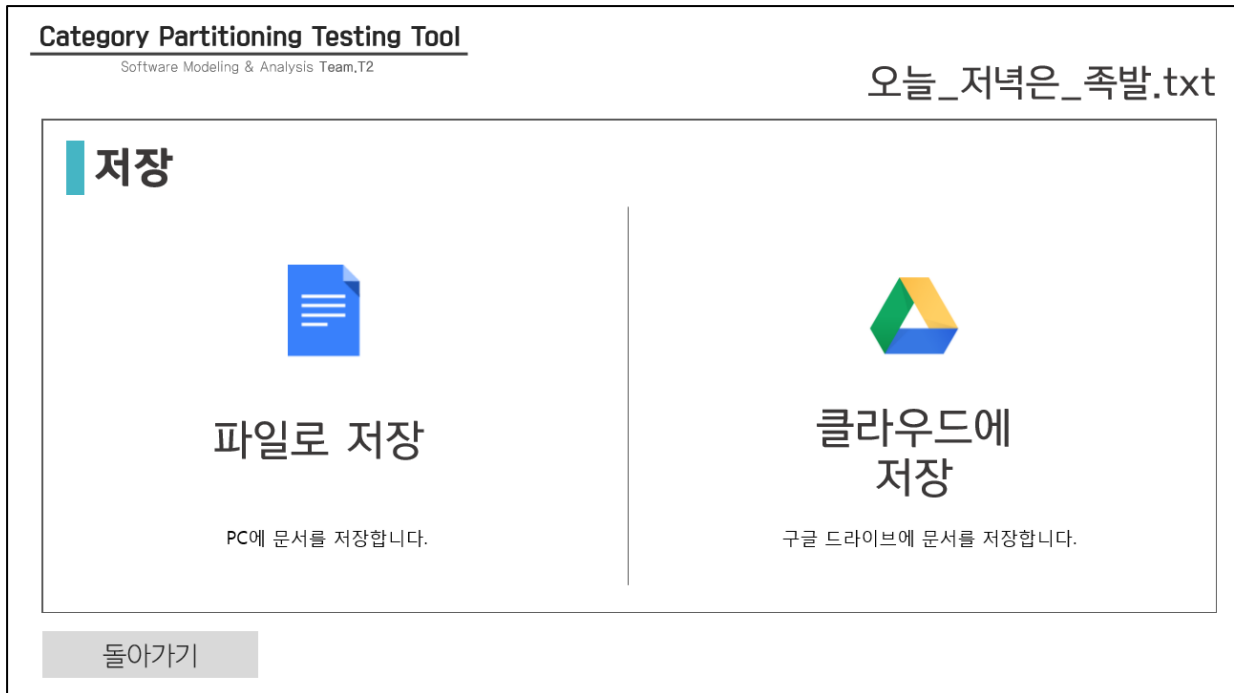
1	6	11			
2	7	12			
3	8	13	18		23
4	9	14	19		24
5	10	15	20		25

이지수 : 룰루
 조만규 : 칼라
 서희진 : 야호
 박성규 : 까
 지이수 : 지슈슈
 미인규 : 밍키
 고양이 : 애옹애옹
 Software : 소모
 Modeling : 파이팅
 Analysis : π

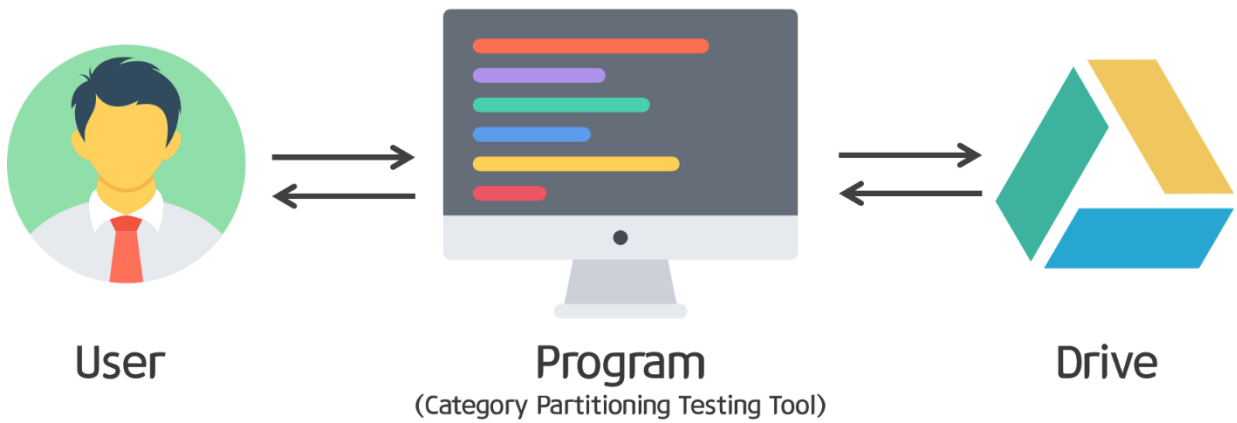
돌아가기

저장하기

5. 파일 저장 화면

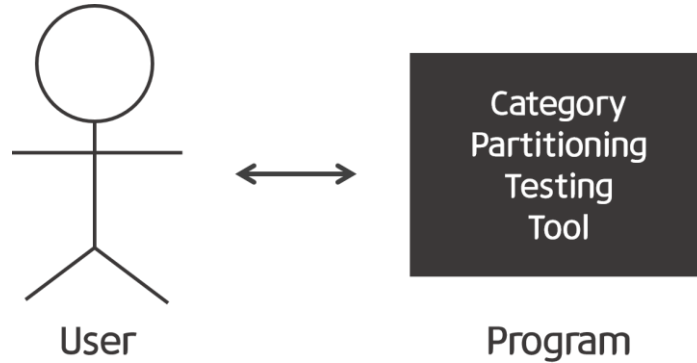


Activity 1006. Define Draft System Architecture



Activity 1007. Define Business Use Case

1. Define System Boundary



2. Identify and Describe Actors

A. User

User 는 다음과 같은 행동을 할 수 있다.

I. 메인 화면

1. txt 확장자의 Test 명세 파일을 컴퓨터 에서 불러올 수 있다.
2. txt 확장자의 Test 명세 파일을 구글 드라이브에서 불러 올 수 있다.
3. 종료 버튼을 눌러서 프로그램을 종료할 수 있다.

II. Category 출력 화면

1. 특정 Category 에 마우스를 올려서 커서가 위치한 Category 의 이름의 색이 변하는 것을 확인한다.
2. Category 의 이름들을 클릭하여 각각의 Category 가 가지고 있는 변수의 출력화면으로 이동한다.
3. 조합하기 버튼을 눌러서 Test Script 화면으로 이동한다. 프로그램을 종료할 수 있다.
4. 로고를 눌러서 메인 화면으로 이동한다.

III. Value 출력화면

1. 각각의 value 이름에 마우스를 올려서 커서가 위치한 value 의 이름의 색이 변하는 것을 확인하고, 각각의 value 의 제약사항을 확인한다.
2. 돌아가기 버튼을 눌러서 category 출력 화면으로 이동한다.
3. 로고를 눌러서 메인 화면으로 이동한다.

IV. Test Script 출력화면

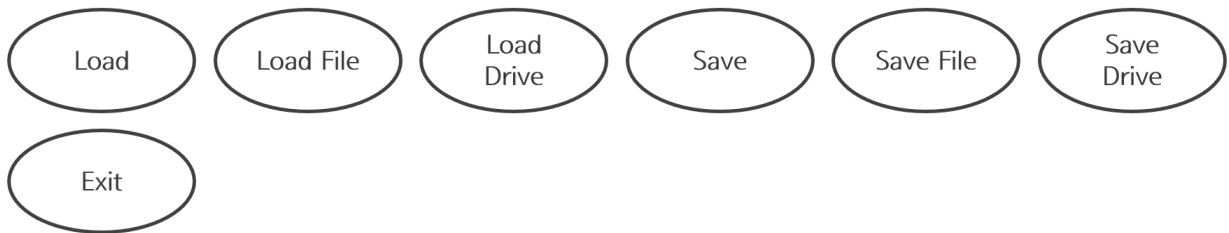
1. 각각의 test case 번호에 마우스를 올려서 커서가 위치한 test case 의 category 와 value 를 확인한다.
2. 돌아가기 버튼을 눌러서 category 출력 화면으로 이동한다.
3. 저장하기 버튼을 눌러서 저장화면으로 이동한다.
4. 로고를 눌러서 메인 화면으로 이동한다.

V. 저장화면

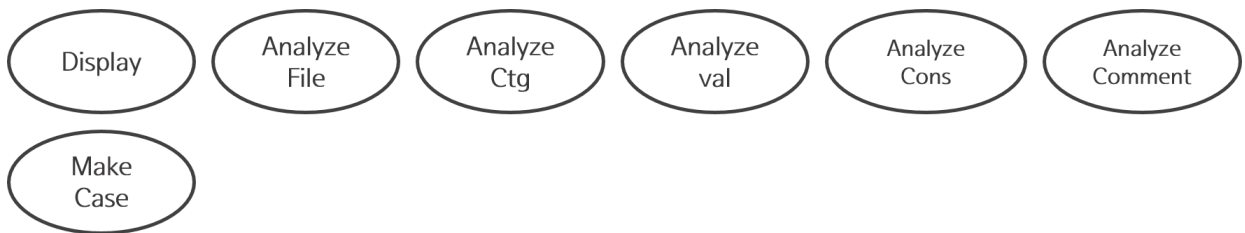
1. 파일로 저장 버튼을 눌러서 PC 에 문서를 저장한다.
2. 클라우드에 저장 버튼을 눌러서 구글 드라이브에 문서를 저장한다.
3. 로고를 눌러서 메인 화면으로 이동한다.

3. Identify Use-Case

A. Use-Cases by Actor-Based



B. Use-Cases by Event-Based



4. Allocate System Functions into Related Use-Cases

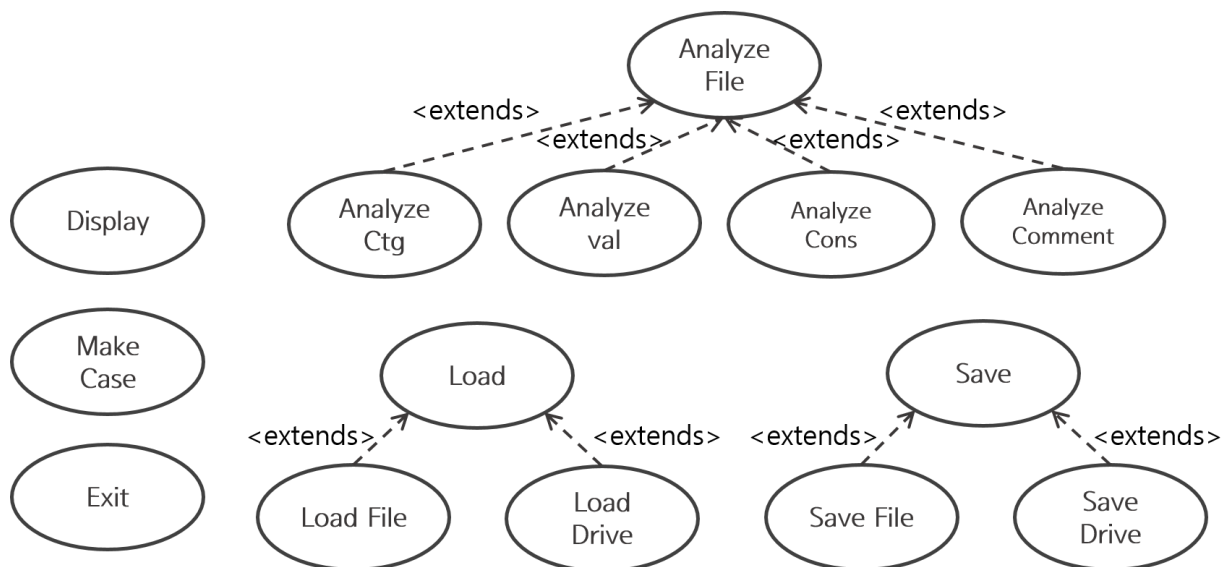
A. Use-Cases by Event-Based

Ref	Function	Use Case
R.1	DisplayMain	DisplayMain
R.2	Init	Init
R.3	LoadFile	LoadFile
R.4	LoadDrive	LoadDrive
R.5	GoHome	GoHome
R.6	Exit	Exit
R.7	GoScript	GoScript
R.7.1	MakeCase	MakeCase
R.8	AnalyzeFile	AnalyzeFile
R.8.1	AnalyzeCtg	AnalyzeCtg
R.8.2	AnalyzeVal	AnalyzeVal
R.8.3	AnalyzeCons	AnalyzeCons
R.8.4	AnalyzeComment	AnalyzeComment
R.9	DisplayCtg	DisplayCtg
R.10	ColoringCtg	ColoringCtg
R.11	GoVal	GoVal
R.11.1	FindVal	FindVal
R.12	DisplayVal	DisplayVal
R.13	ColoringVal	ColoringVal
R.14	ShowCons	ShowCons
R.15	GoCtg	GoCtg
R.16	DisplayScript	DisplayScript
R.17	ShowCase	ShowCase
R.18	ColoringCase	ColoringCase
R.19	GoSave	GoSave
R.20	SaveFile	SaveFile
R.21	SaveDrive	SaveDrive

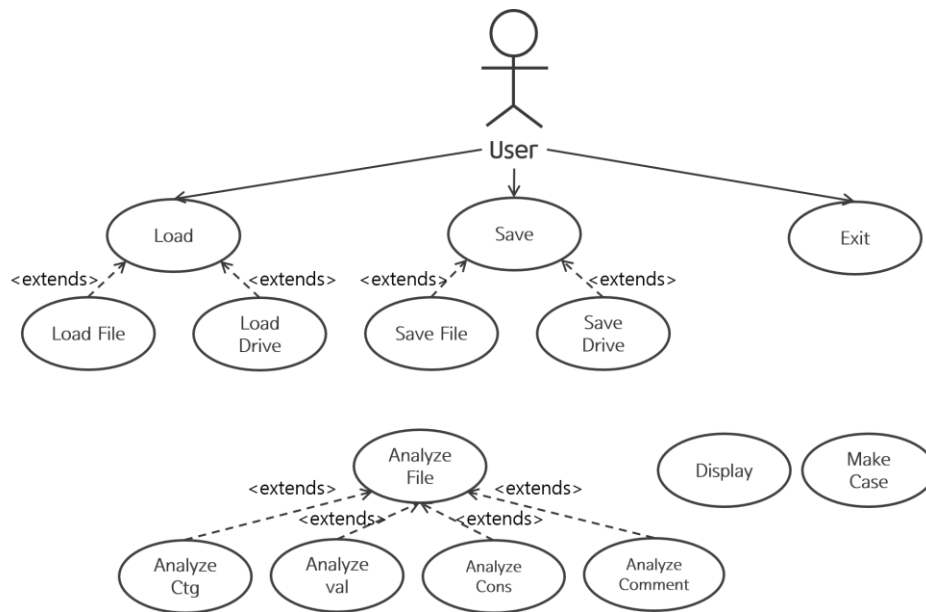
5. Categorize Use-Cases

Use-Case	Category
Load	Primary
LoadFile	Primary
LoadDrive	Primary
AnalyzeFile	Primary
AnalyzeCtg	Primary
AnalyzeVal	Primary
AnalyzeCons	Primary
AnalyzeComment	Primary
MakeCase	Primary
Display	Primary
Save	Primary
SaveFile	Primary
SaveDrive	Primary
Exit	Primary

6. Identify Relationships between Use-Cases



7. Draw a Use-Case Diagram



8. Describe Use-Cases

Use-Case Name	Description
1. Load	파일을 불러온다.
Actor	
User	

Use-Case Name	Description
2. LoadFile	폴더에서 파일을 불러온다.
Actor	
User	

Use-Case Name	Description
3. LoadDrive	Google Drive 에서 파일을 불러온다.
Actor	
User	

Use-Case Name	Description
4. AnalyzeFile	불러온 파일을 프로그램의 형식에 맞춰 분석한다.
Actor	
None	

OOPT Stage 1000 <Plan and Elaboration>

Use-Case Name	Description
5. AnalyzeCtg	불러온 파일에서 Category 를 분석한다.
Actor	
None	

Use-Case Name	Description
6. AnalyzeVal	불러온 파일에서 Value 를 분석한다.
Actor	
None	

Use-Case Name	Description
7. AnalyzeCons	불러온 파일에서 Constraints 를 분석한다.
Actor	
None	

Use-Case Name	Description
8. AnalyzeComment	불러온 파일에서 Comment 를 분석한다.
Actor	
None	

Use-Case Name	Description
9. makeCase	파일에서 분석된 Category 와 Value 와 Constraint 를 기반으로 테스트 케이스를 조합한다.
Actor	
None	

Use-Case Name	Description
10. Display	Category 와 Value 와 Constraints 와 Script 목록을 출력한다.
Actor	
None	

Use-Case Name	Description
11. Save	Test Script 를 저장한다.
Actor	
None	

OOPT Stage 1000 <Plan and Elaboration>

Use-Case Name	Description
12. SaveFile	Test Script 를 컴퓨터에 저장한다.
Acto	
User	

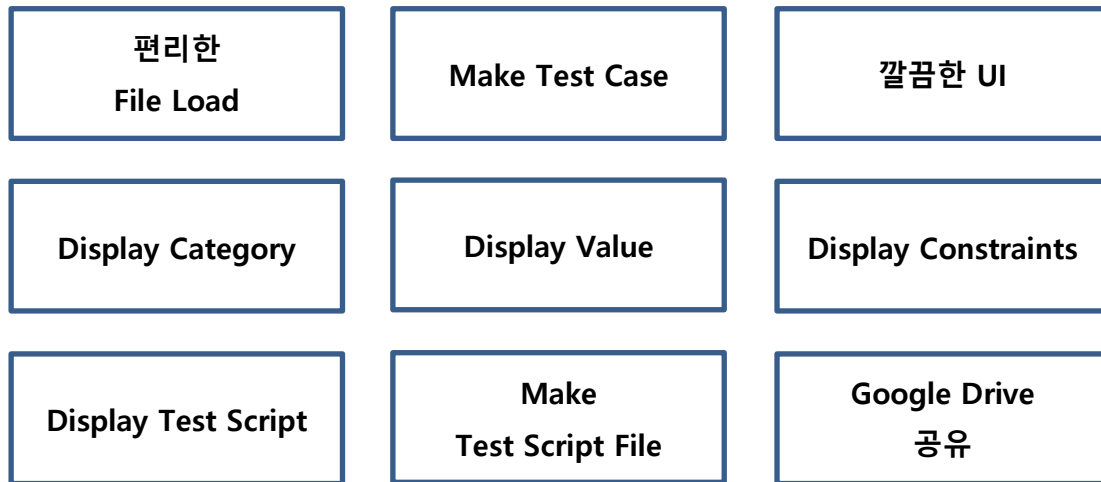
Use-Case Name	Description
13. SaveDrive	Test Script 를 구글드라이브에 저장한다.
Acto	
User	

Use-Case Name	Description
14. Exit	프로그램을 종료한다.
Acto	
User	

9. Rank Use-Cases

Use-Case Number & Name	Rank
Load	High
LoadFile	High
LoadDrive	High
AnalyzeFile	High
AnalyzeCtg	High
AnalyzeVal	High
AnalyzeCons	High
AnalyzeComment	High
MakeCase	High
Display	High
Save	High
SaveFile	High
SaveDrive	High
Exit	High

Activity 1008. Define Business Concept Model



Activity 1009. Define System Test Case

Ref	Function	Test Case
R.1	Load	파일을 정상적으로 불러오는지 확인한다.
R.1.1	LoadFile	폴더 지정 화면이 나타나고, 정상적으로 지정되어 파일을 정상적으로 불러오는지 확인한다.
R.1.2	LoadDrive	Google Drive 로그인 화면이 나타나고, 파일이 정상적으로 지정되어 파일을 정상적으로 불러오는지 확인한다.
R.2	AnalyzeFile	불러온 파일을 분석하여 Test Case 조합에 필요한 정보로 가공한다.
R.2.1	AnalyzeCtg	불러온 파일에서 Category 를 추출하여 프로그램 내부에 정상적으로 저장되는지 확인한다.
R.2.2	AnalyzeVal	불러온 파일에서 Value 를 추출하여 프로그램 내부에 정상적으로 저장되는지 확인한다.
R.2.3	AnalyzeCons	불러온 파일에서 Constraints 를 추출하여 프로그램 내부에 정상적으로 저장되는지 확인한다.
R.2.4	AnalyzeComment	불러온 파일에서 Comment 를 추출하여 프로그램 내부에 정상적으로 저장되는지 확인한다.
R.3	MakeCase	프로그램 내부에 저장된 Category, Value, Constraint 로 Test Case 를 정상적으로 조합하는지 확인한다.
R.4	Display	불러온 파일에서 추출하여 프로그램 내부에 저장된 Category, Value, Constraints, 조합된 Test Case 의 목록을 정상적으로 출력하는지 확인한다.
R.5	Save	Test Script 가 정상적으로 저장되는지 확인한다.
R.5.1	SaveFile	Test Script 가 정상적으로 txt 파일로 저장되는지 확인한다.
R.5.2	SaveDrive	Test Script 가 정상적으로 Google Drive 에 업로드 되는지 확인한다.
R.6	Exit	프로그램이 정상적으로 종료되는지 확인한다.

Activity 1010. Refine Plan

1. Project Scope

너랑나 Category Partitioning Tool

기존의 Category Partitioning Testing Tool 에 구글 드라이브를 연동하여 팀원들끼리 문서를 실시간으로 공유할 수 있게 하고, 깔끔한 UI 를 활용하여 조합된 테스트케이스를 한눈에 알아볼 수 있게 하는 것을 목표로 한다.

2. Project Objectives

-수동으로 Test Scripts 를 만드는 것은 많은 시간이 걸리고 비효율적인 일이기 때문에 Test Specification 을 기준으로 Test Scripts 를 만들어주는 Category Partitioning Testing Tool 을 개발한다.

-프로젝트를 팀으로 진행하는 경우가 많기 때문에 팀 간의 빠른 문서 공유를 위해 구글드라이브와 연동한다.

3. Functional Requirements

- Load
- LoadFile
- LoadDrive
- AnalyzeFile
- AnalyzeCategory
- AnalyzeValue
- AnalyzeConstraints
- AnalyzeComment
- MakeCase
- Display
- Save
- SaveFile
- SaveDrive
- Exit

4. Performance Requirements

- A. 직관적인 인터페이스
- B. 텍스트 파일로 작성된 Test Specification 만 분석한다.

5. Operating Environment

Microsoft Windows 7 이상

6. User Interface Requirements

알아보기 쉬운 Interface 를 통해 특별한 설명 없이 프로그램을 수행하고, 간결한 결과 화면을 통해 사용자에게 결과를 정확하게 전달한다.

7. Other Requirements

Google 계정을 필요로 한다.

8. Resource Estimation

- A. Human Resource : 4 명
- B. Project Duration : 3 개월(12 주)
- C. Human Efforts(Man-Month) : 12
- D. Cost : 식대 1,200,000 원 (5,000 원/일 × 4 명 × 5 일/주 × 12 주)

OOPT Stage 1000 <Plan and Elaboration>

9. Scheduling

Stage	Phase/Activity	Schedule (Week)															
		1	2	3	4	5	6	7	8	9	10	11	12				
1000 Plan & Elaborate	1001. Define Draft Plan	█															
	1002. Create Preliminary Investigation Report	█															
	1003. Define Requirements	█															
	1004. Record Terms in Glossary		█														
	1005. Implement Prototype		█														
	1006. Define Draft System Architecture		█														
	1007. Define Business Use Case		█														
	1008. Define Business Concept Model		█	█													
	1009. Define System Test Case		█	█													
	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 Class 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															█	