



2016 소프트웨어 모델링&분석

# 그놈! Clone Checker

Team 4





# Contents

1. Revise Plan
2. Define Essential Use Cases
3. Define Domain Model
4. Refine Glossary
5. Define System Sequence Diagrams
6. Define Operation Contracts
7. Refine System Test Case
8. Analyze Traceability Analysis



# 1. Revise Plan

# Revise Plan

OSP Stage 1000 ver1	OSP Stage 1000 ver 2, ver 3
유사도 기반 중심 파일 예측 Clone Checker	그 놈! Clone Checker

(수정)

간결해진 프로젝트 명

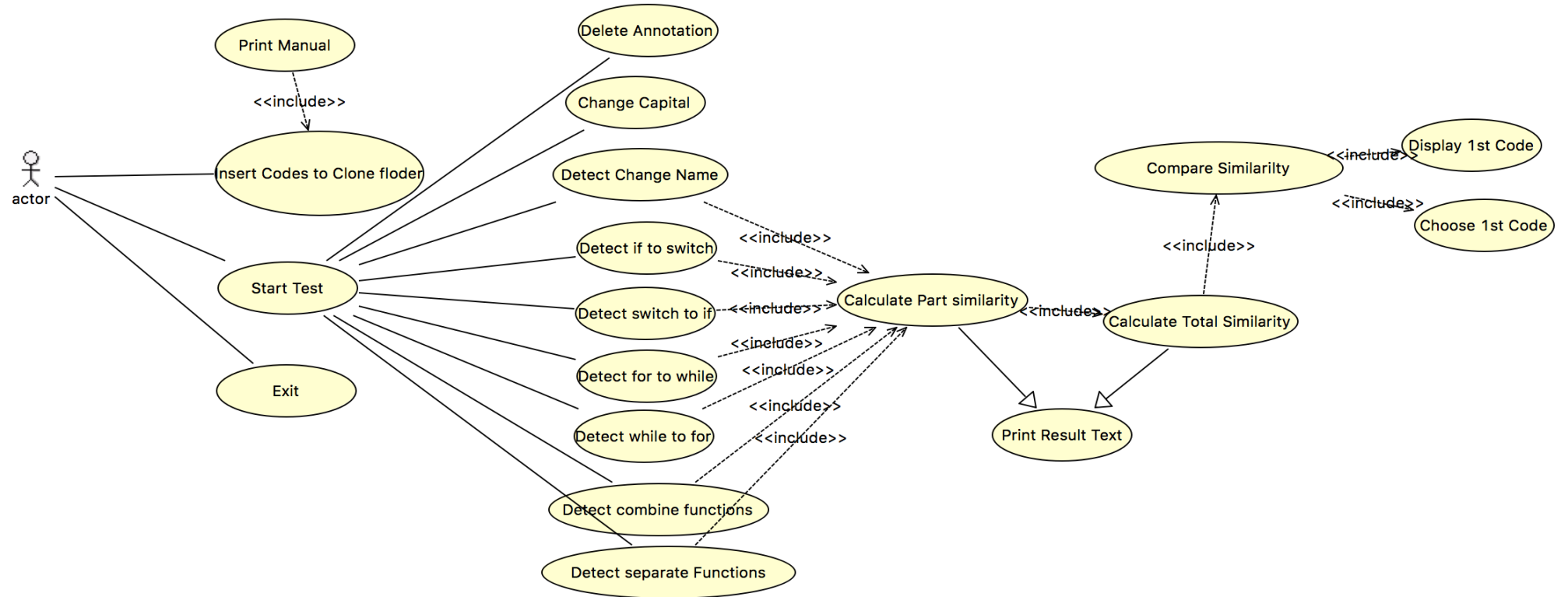
유사도 기반 중심 파일 예측 Clone Checker -> 그 놈! Clone Checker

# Revise Plan

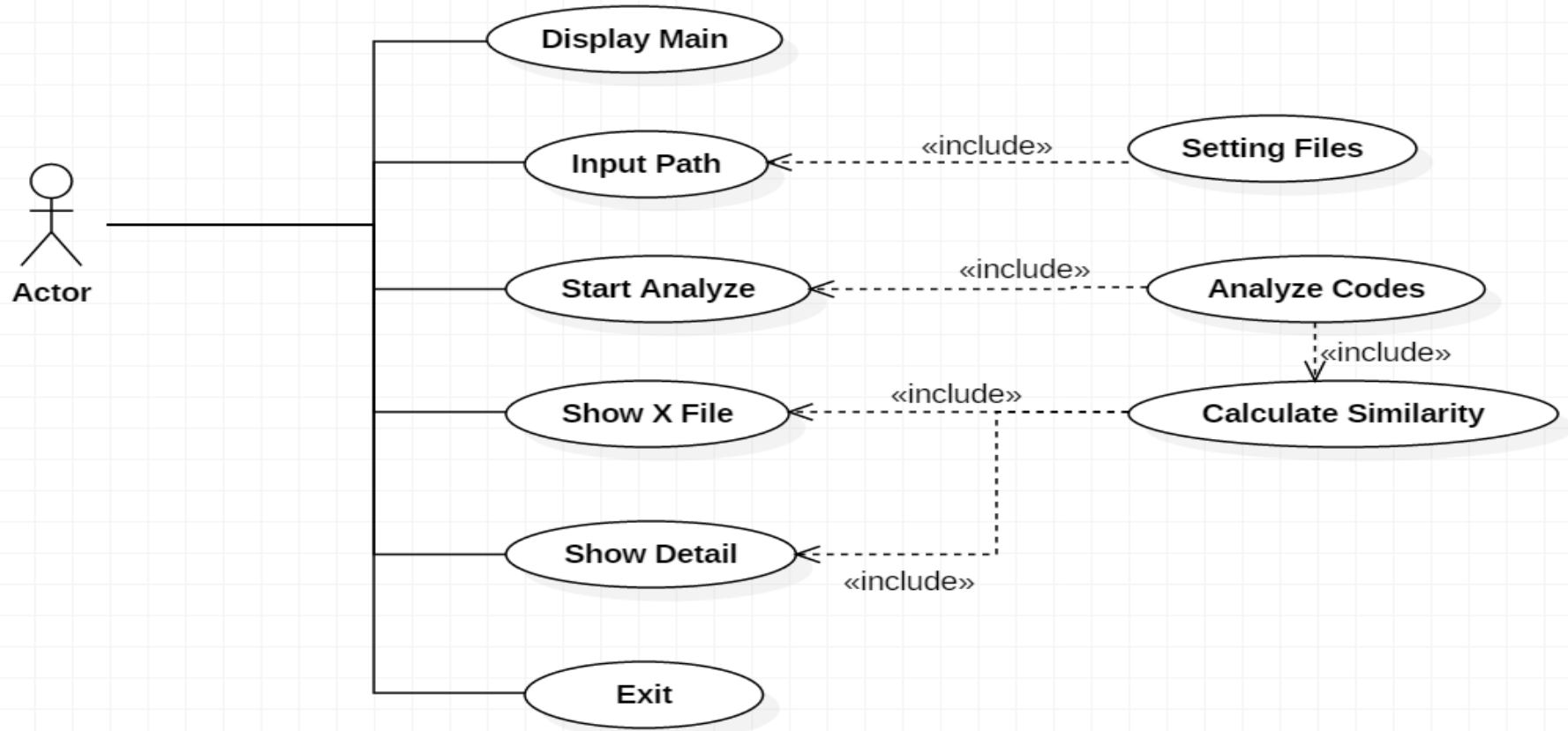
OSP Stage 1000 ver1, ver2	OSP Stage 1000 ver3
Print Manual	Display Main
Start Test	Input Path
Delete Annotation	Load Files
Change Capital	Delete Annotation
Detect Change Name	Change Capital
Detect for To while	Start Analyze
Detect while To for	Analyze Change Name
Detect if To switch	Analyze for To while
Detect switch To if	Analyze while To for
Detect Separate Functions	Analyze if To switch
Detect Combine Functions	Analyze switch To if
Calculate Part Similarity	Analyze Separate Functions
Calculate Total Similarity	Analyze Combine Functions
Write Result Text	Calculate Similarity
Choose 1 <sup>st</sup> Code	Save Detail
Display 1 <sup>st</sup> Code	Find X File
Exit	Show X File
	Show Detail
	Exit

(수정)  
세분화된 Functional  
Requirements

# Revise Plan



# Revise Plan





## 2. Define Essential Use Cases



# Define Essential Use Cases

Use Case	Display Main
Actor	User
Purpose	프로그램을 시작한다.
Overview	프로그램의 첫 메인 화면을 보여준다.
Type	Primary and Essential
Cross Reference	System Function : R 1.1
Pre-Requisites	N/A
Typical Courses of Events	(A) : Actor, (S):System 1. (A) 사용자가 프로그램을 실행한다. 2. (S) 프로그램의 메인 UI를 띄운다.
Alternative Courses of Events	N/A
Exceptional Courses of Events	N/A

# Define Essential Use Cases

Use Case	Input Path
Actor	User
Purpose	유사도 검사를 진행할 대상의 위치 파악.
Overview	유사도 검사를 할 폴더의 경로를 입력한다.
Type	Primary and Essential
Cross Reference	Use Case : Setting Files System Function : R 1.2, R 1.3, R 1.3.1, R 1.3.2
Pre-Requisites	N/A
Typical Courses of Events	(A) : Actor, (S):System 1. (A) 사용자가 유사도 검사를 진행할 소스코드들이 있는 폴더의 경로를 입력한 뒤, 입력 버튼을 누른다. 2. (S) 해당 경로에 위치한 폴더의 파일 리스트를 읽어온다. 3. (S) 해당 폴더의 모든 소스파일의 주석을 제거한다. 4. (S) 해당 폴더의 모든 소스파일의 대문자를 소문자로 통합한다.
Alternative Courses of Events	N/A
Exceptional Courses of Events	N/A

# Define Essential Use Cases

Use Case	Setting Files
Actor	System
Purpose	소스코드 파일 정리
Overview	소스코드의 주석을 제거하고, 소문자로 통일한다.
Type	Primary and Essential
Cross Reference	System Function : R 1.3, R 1.3.1, R 1.3.2
Pre-Requisites	N/A
Typical Courses of Events	(A) : Actor, (S):System 1. (S) 소스코드의 모든 주석을 제거한다. 2. (S) 소스코드의 모든 문자를 소문자로 통일한다.
Alternative Courses of Events	N/A
Exceptional Courses of Events	N/A

# Define Essential Use Cases

Use Case	Start Analyze
Actor	User
Purpose	유사도 검사 시작
Overview	유사도 검사를 시작한다.
Type	Primary and Essential
Cross Reference	Use Case : Analyze Codes System Function : R 2.1, R 2.2, R 2.2.1, R 2.2.2, R 2.2.3, R 2.2.4, R 2.2.5, R 2.2.6
Pre-Requisites	N/A
Typical Courses of Events	(A) : Actor, (S):System 1. (A) 사용자가 Start 버튼을 누른다. 2. (S) 두 소스코드 사이의 변수, 함수, 반복문, 조건문에 있어서 얼마나 유사한지 분석한다.
Alternative Courses of Events	N/A
Exceptional Courses of Events	N/A

# Define Essential Use Cases

Use Case	Analyze Codes
Actor	System
Purpose	소스코드의 분석
Overview	소스코드 사이의 변수, 함수, 반복문, 조건문 유사도를 각각 분석한다.
Type	Primary and Essential
Cross Reference	System Function : R 2.2, R 2.2.1, R 2.2.2, R 2.2.3, R 2.2.4, R 2.2.5, R 2.2.6
Pre-Requisites	N/A
Typical Courses of Events	(A) : Actor, (S):System 1. (S) 소스코드의 변수, 함수, 반복문, 조건문을 분석한다. 2. (S) 변수명과 함수명을 바꾸었는지 분석한다. 3. (S) for문을 while문으로 바꾸었는지 분석한다. 4. (S) while문을 for문으로 바꾸었는지 분석한다. 5. (S) if문을 switch문으로 바꾸었는지 분석한다. 6. (S) switch문을 if문으로 바꾸었는지 분석한다. 7. (S) 함수를 의도적으로 나누었는지 분석한다. 8. (S) 함수를 의도적으로 합쳤는지 분석한다.
Alternative Courses of Events	N/A
Exceptional Courses of Events	N/A

# Define Essential Use Cases

Use Case	Calculate Similarity
Actor	System
Purpose	유사도의 계산 및 그놈 파일 찾기
Overview	코드 분석을 바탕으로 유사도를 계산하여 저장하고 이를 바탕으로 중심이 되는 그놈 파일을 찾는다.
Type	Primary and Essential
Cross Reference	Use Case : Analyze Code System Function : R 2.2, R 2.2.1, R 2.2.2, R 2.2.3, R 2.2.4, R 2.2.5, R 2.2.6, R 3.1, R 3.1.1, R 3.1.2
Pre-Requisites	N/A
Typical Courses of Events	(A) : Actor, (S):System 1. (S) 소스코드 사이의 변수, 함수, 반복문, 조건문 유사도를 각각 분석한다. 2. (S) 분석한 내용을 바탕으로 가중치를 적용하여 유사도를 계산한다. 3. (S) 유사도 검사의 내용과 최종 유사도를 각각 txt 파일에 저장한다. 4. (S) 모든 파일의 검사가 끝났으면, 검사 결과의 모든 유사도를 바탕으로 가장 중심이 되는 그놈 파일을 찾는다.
Alternative Courses of Events	N/A
Exceptional Courses of Events	N/A

# Define Essential Use Cases

Use Case	Show X File
Actor	User
Purpose	그놈! 파일 확인
Overview	그놈! 파일의 파일명을 본다.
Type	Primary and Essential
Cross Reference	Use Case : Calculate Similarity System Function : R 3.1, R 3.1.1, R 3.1.2, R 4.1
Pre-Requisites	N/A
Typical Courses of Events	(A) : Actor, (S):System 1. (A) 사용자가 Show X File 버튼을 누른다. 2. (S) 그놈파일의 파일명을 출력한다.
Alternative Courses of Events	N/A
Exceptional Courses of Events	N/A

# Define Essential Use Cases

Use Case	Show Detail
Actor	User
Purpose	각각의 유사도 검사 결과 확인
Overview	각각의 유사도 검사 결과를 확인한다.
Type	Primary and Essential
Cross Reference	Use Case : Calculate Similarity System Function : R 3.1, R 3.1.1, R 3.1.2, R 4.2
Pre-Requisites	N/A
Typical Courses of Events	(A) : Actor, (S):System 1. (A) 사용자가 Show Detail 버튼을 누른다. 2. (S) 유사도 검사의 기록이 담긴 txt 파일을 보여준다.
Alternative Courses of Events	N/A
Exceptional Courses of Events	N/A



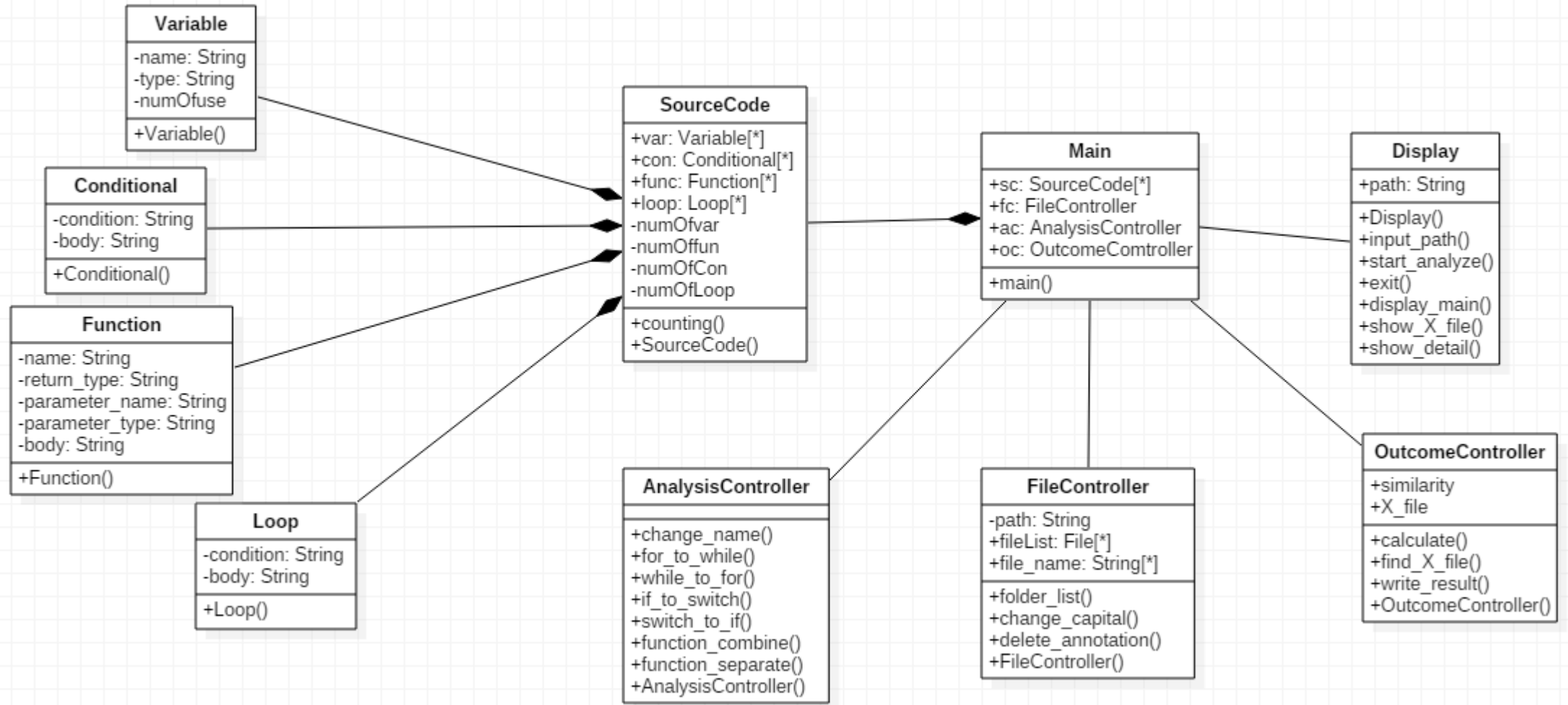
# Define Essential Use Cases

Use Case	Exit
Actor	User
Purpose	프로그램 종료
Overview	프로그램을 종료한다.
Type	Primary and Essential
Cross Reference	System Function : R 5.1
Pre-Requisites	N/A
Typical Courses of Events	(A) : Actor, (S):System 1. (A) 사용자가 Exit 버튼을 누른다. 2. (S) 프로그램을 종료한다.
Alternative Courses of Events	N/A
Exceptional Courses of Events	N/A



# 3. Define Domain Model

# Define Domain Model





# 4. Refine Glossary

# Refine Glossary

Term	Category	Comment
SourceCode	class	코드의 객체
Varialbe	class	변수의 객체
Conditional	class	조건문의 객체
Function	class	함수의 객체
Loop	class	반복문의 객체
Main	class	메인
Display	class	UI 객체
AnalysisController	class	분석을 관리하는 객체

# Refine Glossary

Term	Category	Comment
FileController	class	파일을 관리하는 객체
OutcomeController	class	결과를 관리하는 객체
SourceCode.var[]	attribute	코드가 가지는 변수의 객체
SourceCode.con[]	attribute	코드가 가지는 조건문의 객체
SourceCode.func[]	attribute	코드가 가지는 함수의 객체
SourceCode.loop[]	attribute	코드가 가지는 반복문의 객체
SourceCode.numOfvar	attribute	코드에 있는 변수의 수
SourceCode.numOffun	attribute	코드에 있는 함수의 수

# Refine Glossary

Term	Category	Comment
SourceCode.numOfcon	attribute	코드에 있는 조건문의 수
SourceCode.numOfloop	attribute	코드에 있는 반복문의 수
Variable.name	attribute	변수의 이름
Variable.type	attribute	변수의 자료형
Variable.numOfuse	attribute	변수의 사용횟수
Conditional.condition	attribute	조건문의 조건
Conditional.body	attribute	조건문의 내용
Function.name	attribute	함수의 이름

# Refine Glossary

Term	Category	Comment
Function.return_type	attribute	함수의 반환형
Function.parameter_name	attribute	함수의 매개변수 이름
Function.parameter_type	attribute	함수의 매개변수 자료형
Function.body	attribute	함수의 내용
Loop.condition	attribute	반복문의 조건
Loop.body	attribute	반복문의 내용
Main.sc[]	attribute	메인에 생성된 SourceCode 객체
Main.fc	attribute	메인에 생성된 FileController 객체



# Refine Glossary

Term	Category	Comment
Main.ac	attribute	메인에 생성된 AnalysisController 객체
Main.oc	attribute	메인에 생성된 OutcomeController 객체
Display.path	attribute	검사할 파일을 담은 폴더의 경로
FileController.Path	attribute	검사할 파일을 담은 폴더의 경로
FileController.fileList	attribute	파일들의 리스트
FileController.file_name	attribute	파일들의 이름
OutcomeController.similarity	attribute	최종 유사도
OutcomeController.X_file	attribute	원본파일로 생각되는 파일

# Refine Glossary

Term	Category	Comment
Display.Display	operation	Display를 초기화 한다
Display.typing_path	operation	파일의 경로를 입력한다.
Display.start_test	operation	검사를 시작한다.
Display.exit	operation	검사를 종료한다.
Main.main	operation	메인을 시작한다.
Variable.Variable	operation	Variable을 초기화 한다.
Conditional.Conditional	operation	Conditional 을 초기화 한다.
Function.Function	operation	Function 을 초기화 한다.

# Refine Glossary

Term	Category	Comment
Loop.Loop	operation	Loop 을 초기화 한다.
SourceCode.counting	operation	numOfvar, numOffun, numOfcon, numOfloop 를 계산한다.
SourceCode.SourceCode	operation	SourceCode 를 초기화 한다.
AnalysisController.change_name	operation	변수 와 함수의 이름이 바뀐곳을 찾는다.
AnalysisController.for_to_while	operation	for 문을 while 문으로 바뀐 곳을 찾는다.
AnalysisController.while_to_for	operation	while 문을 for 문으로 바뀐 곳을 찾는다.
AnalysisController.if_to_switch	operation	if 문을 switch 문으로 바뀐 곳을 찾는다.
AnalysisController.switch_to_if	operation	switch 문을 if 문으로 바뀐 곳을 찾는다.

# Refine Glossary

Term	Category	Comment
AnalysisController.function_combine	operation	함수를 의도적으로 합친 곳을 찾는다
AnalysisController.function_separate	operation	함수를 의도적으로 나눈 곳을 찾는다
AnalysisController.AnalysisController	operation	AnalysisController 를 초기화 한다.
FileController.flober_list	operation	폴더에 있는 파일들을 저장한다.
FileController.change_capital	operation	파일들에 있는 대문자를 소문자로 바꿔준다.
FileController.delete_annotation	operation	파일들에 있는 주석들을 모두 제거한다.
FileController.FileController	operation	FileController 를 초기화 한다.
OutcomeController.calculate	operation	분석을 토대로 결과를 계산한다

# Refine Glossary

Term	Category	Comment
OutcomeController.find_X_file	operation	계산결과를 토대로 원본에 가까운 것 같은 파일을 찾는다.
OutcomeController.write_result	operation	분석결과를 .txt파일로 저장한다.
OutcomeController.OutcomeController	operation	OutcomeController 를 초기화 한다.

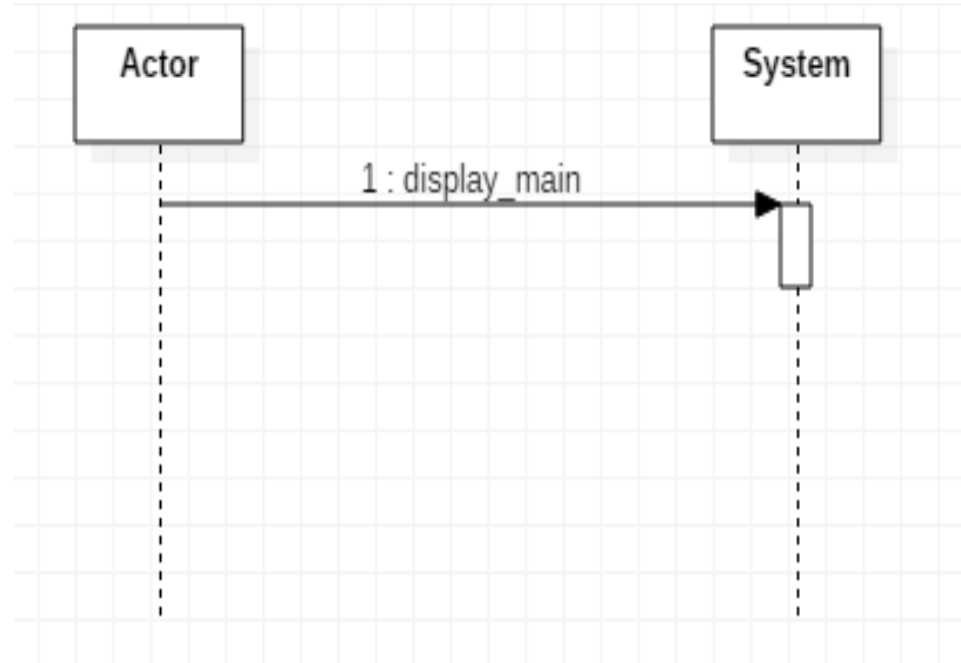


## 5. Define System Sequence Diagrams

# Define System Sequence Diagrams

## 1. Use Case : Display Main

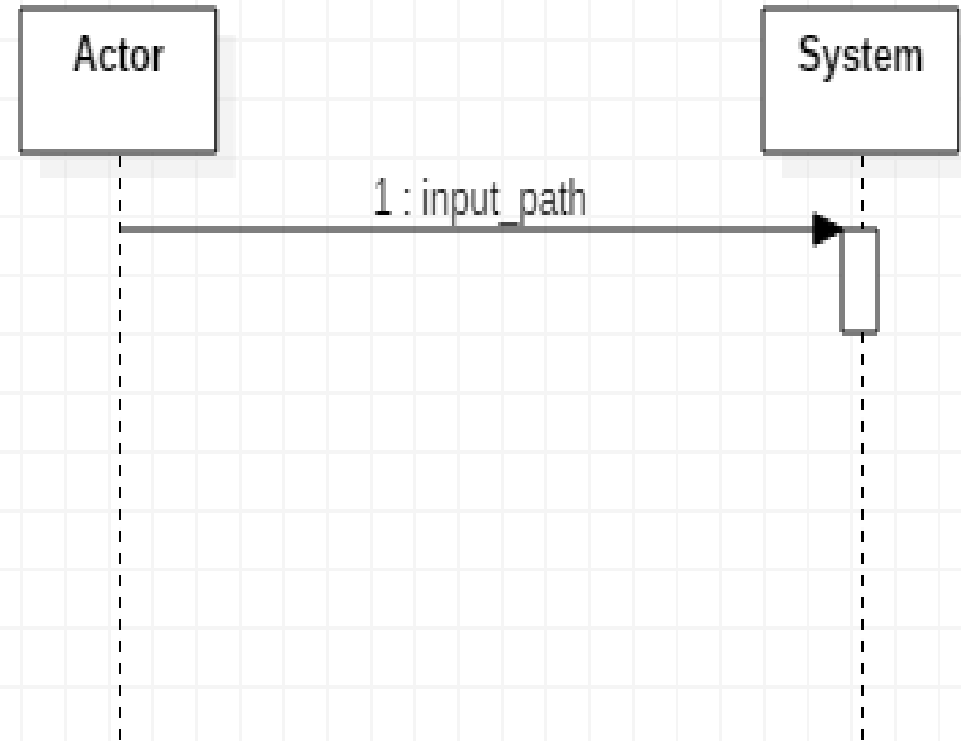
1. User가 프로그램을 실행한다.
2. 프로그램의 메인 UI를 띄운다.



# Define System Sequence Diagrams

## 2. Use Case : Input Path

1. 사용자가 유사도 검사를 진행할 소스코드들이 있는 폴더의 경로를 입력한 뒤, 입력 버튼을 누른다.
2. 해당 경로에 위치한 폴더의 파일 리스트를 읽어온다.
3. 해당 폴더의 모든 소스파일의 주석을 제거한다.
4. 해당 폴더의 모든 소스파일의 대문자를 소문자로 통합한다.

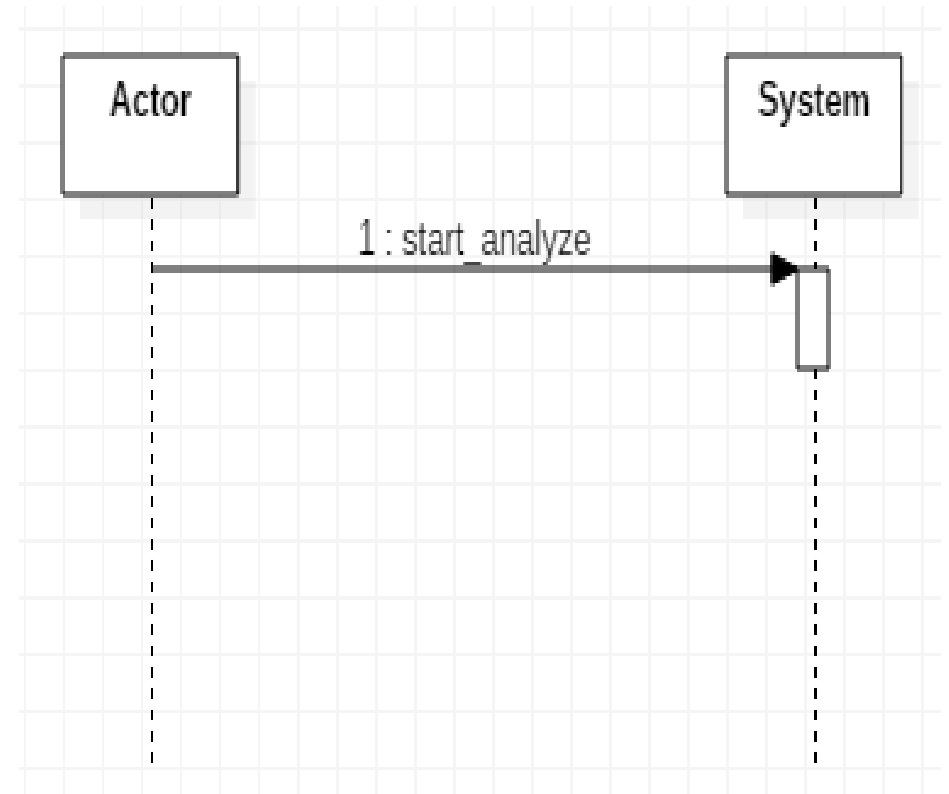




# Define System Sequence Diagrams

## 3. Use Case : Start Analyze

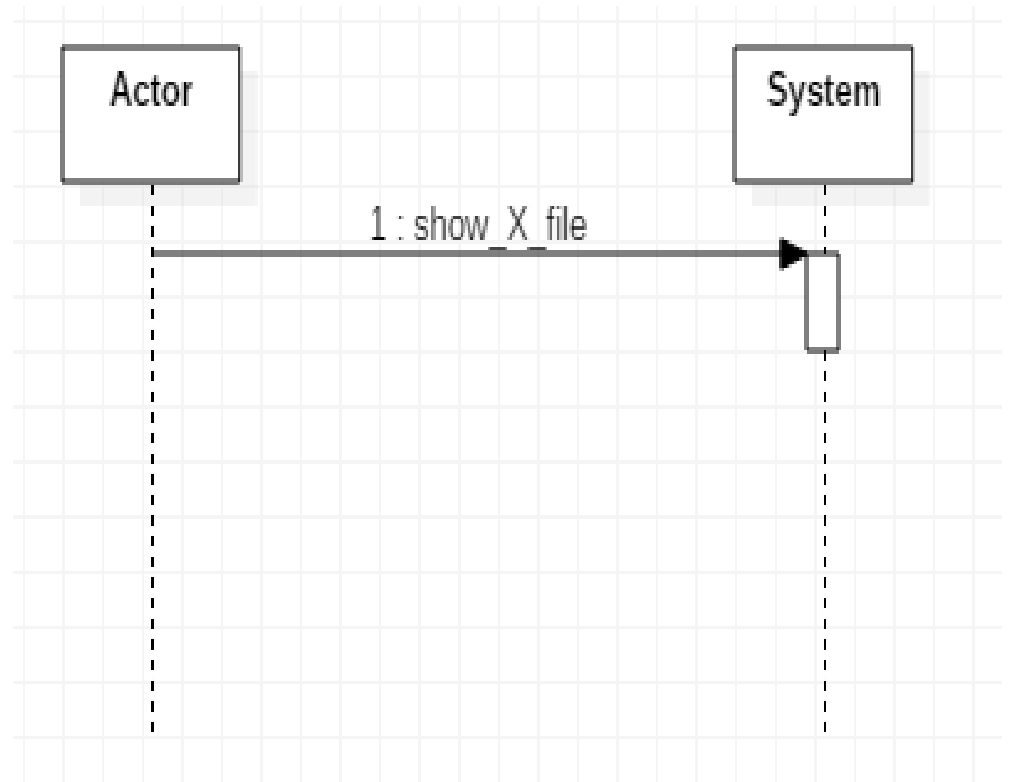
1. 사용자가 Start 버튼을 누른다.
2. 두 소스코드 사이의 변수, 함수, 반복문, 조건문에 있어서 얼마나 유사한지 분석한다.



# Define System Sequence Diagrams

## 4. Use Case : Show X File

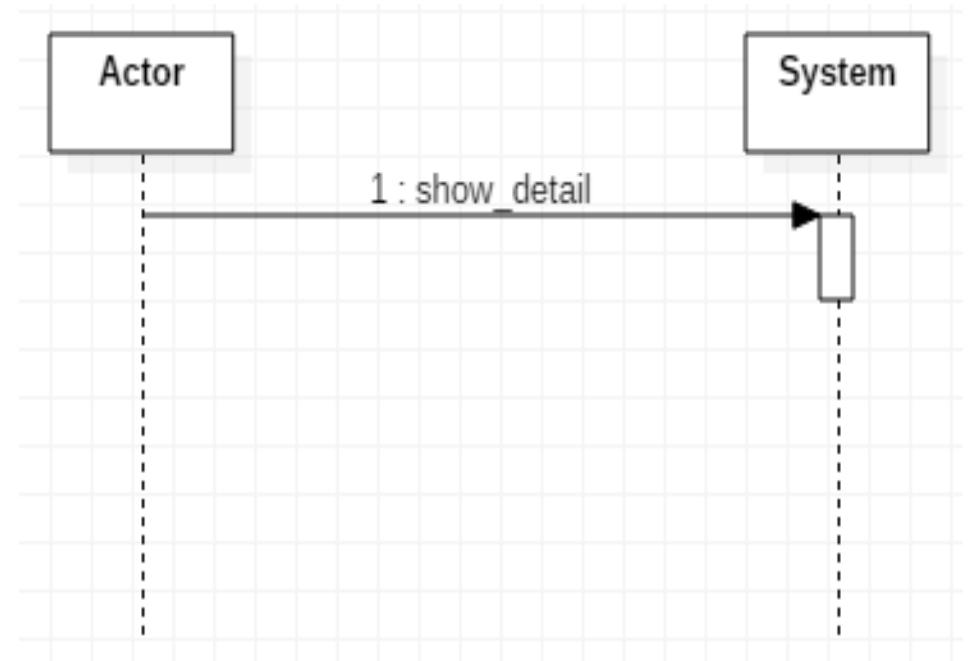
1. 사용자가 Show X File 버튼을 누른다.
2. 그놈파일의 파일명을 출력한다.



# Define System Sequence Diagrams

## 5. Use Case : Show Detail

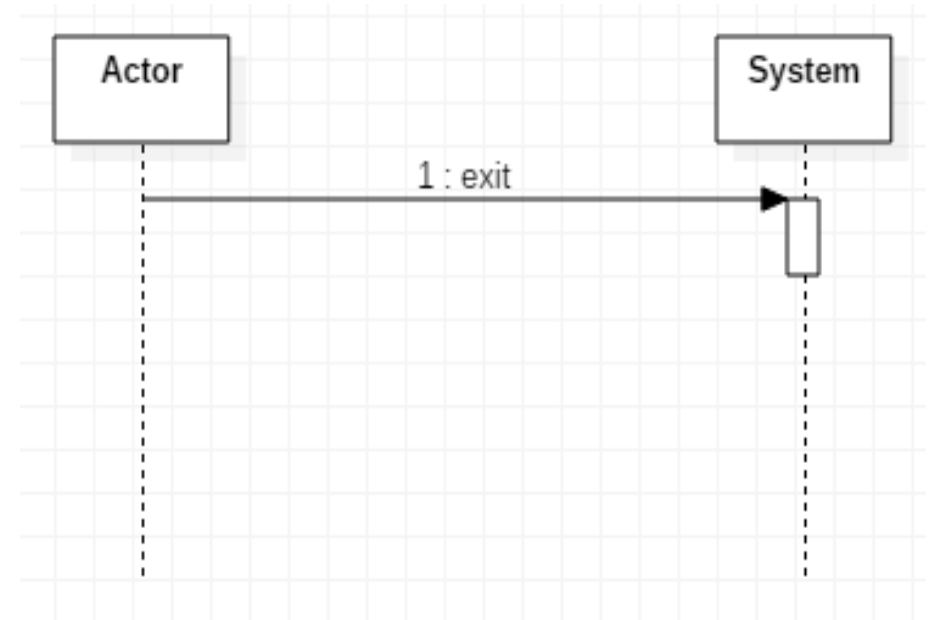
1. 사용자가 Show Detail 버튼을 누른다.
2. 유사도 검사의 기록이 담긴 txt 파일을 보여준다.



# Define System Sequence Diagrams

## 6. Use Case : Exit

1. 사용자가 Exit 버튼을 누른다.
2. 프로그램을 종료한다.





# 6. Define Operation Contracts

# Define Operation Contracts

Use Case	Name of Actor-Activated Event	System Operations
Display Main	Display Main()	display_main()
Input Path	Input Path()	input_path()
Start Analyze	Start Analyze()	start_anlayze()
Show X File	Show X File()	show_X_file()
Show Detail	Show Detail()	show_detail()
Exit	Exit()	exit()

# Define Operation Contracts

<b>Name</b>	display_main()
<b>Responsibilities</b>	UI 를 포함한 실행 초기 화면을 띄운다.
<b>Type</b>	System
<b>Cross Reference</b>	System Function : R 1.1
<b>Exception</b>	N/A
<b>Output</b>	입력, Show X File, Show Detail, Start, Exit 버튼을 포함한 UI 를 출력한다.
<b>Pre-Conditions</b>	N/A
<b>Post-Conditions</b>	주 메뉴 화면

# Define Operation Contracts

<b>Name</b>	input_path()
<b>Responsibilities</b>	사용자에게 소스코드들이 들어있는 폴더의 경로를 입력받는다.
<b>Type</b>	System
<b>Cross Reference</b>	Use Case : Setting Files System Function : R 1.2, R 1.3, R 1.3.1, R 1.3.2
<b>Exception</b>	N/A
<b>Output</b>	해당 경로에 위치한 폴더의 파일 리스트를 읽어온다. 해당 폴더의 모든 소스파일의 주석을 제거한다. 해당 폴더의 모든 소스파일의 대문자를 소문자로 통합한다.
<b>Pre-Conditions</b>	User 가 폴더 경로를 입력해야 한다.
<b>Post-Conditions</b>	N/A



# Define Operation Contracts

Name	start_analyze()
Responsibilities	유사도 검사를 시작한다.
Type	System
Cross Reference	Use Case : Analyze Codes System Function : R 2.1, R 2.2, R 2.2.1, R 2.2.2, R 2.2.3, R 2.2.4, R 2.2.5, R 2.2.6
Exception	N/A
Output	사용자가 Start 버튼을 누른다. 두 소스코드 사이의 변수, 함수, 반복문, 조건문에 있어서 얼마나 유사한지 분석한다.
Pre-Conditions	N/A
Post-Conditions	

# Define Operation Contracts

<b>Name</b>	show_X_file()
<b>Responsibilities</b>	그놈 파일의 파일명을 출력한다.
<b>Type</b>	System
<b>Cross Reference</b>	Use Case : Calculate Similarity System Function : R 3.1, R 3.1.1, R 3.1.2, R 4.1
<b>Exception</b>	N/A
<b>Output</b>	txt 파일에 저장되어 있는 유사도를 바탕으로 가장 중심이 되는 파일(그놈 파일) 을 찾는다. 그놈파일의 파일명을 출력한다.
<b>Pre-Conditions</b>	N/A
<b>Post-Conditions</b>	N/A

# Define Operation Contracts

<b>Name</b>	show_detail()
<b>Responsibilities</b>	유사도 검사의 내용을 보여준다.
<b>Type</b>	System
<b>Cross Reference</b>	Use Case : Calculate Similarity System Function : R 3.1, R 3.1.1, R 3.1.2, R 4.2
<b>Exception</b>	N/A
<b>Output</b>	유사도 검사의 내용이 담긴 txt 파일을 보여준다.
<b>Pre-Conditions</b>	N/A
<b>Post-Conditions</b>	N/A

# Define Operation Contracts

<b>Name</b>	exit()
<b>Responsibilities</b>	그놈! Clone Checker 를 종료한다
<b>Type</b>	System
<b>Cross Reference</b>	System Fucntion : R 5.1 Use Case : Exit
<b>Exception</b>	N/A
<b>Output</b>	프로그램을 종료한다.
<b>Pre-Conditions</b>	N/A
<b>Post-Conditions</b>	N/A



# 7. Refine System Test Case

# Refine System Test Case

Identifier ↴	Feature ↴
CK.STC.100 ↴	Main 화면이 제대로 실행이 되었는지 검사한다. ↴
CK.STC.101 ↴	사용자가 입력한 경로가 정확하게 입력되었는지 검사한다. ↴
CK.STC.102 ↴	사용자가 입력한 경로에 파일들이 잘 들어있는지 검사한다. ↴
CK.STC.103 ↴	사용자가 입력한 경로에 파일들을 잘 불러왔는지 검사한다. ↴
CK.STC.110 ↴	파일 내의 모든 주석이 지워졌는지 검사한다. ↴
CK.STC.112 ↴	파일 내의 모든 문자가 소문자로 바뀌었는지 검사한다. ↴
CK.STC.200 ↴	분석을 잘 시작하는지 검사한다. ↴
CK.STC.210 ↴	변수 명을 바꾼 코드를 잘 찾아 냈는지 검사한다. ↴
CK.STC.211 ↴	함수 명을 바꾼 코드를 잘 찾아 냈는지 검사한다. ↴
CK.STC.212 ↴	if 문을 switch 문으로 바꾼 코드를 잘 찾아 냈는지 검사한다. ↴

# Refine System Test Case

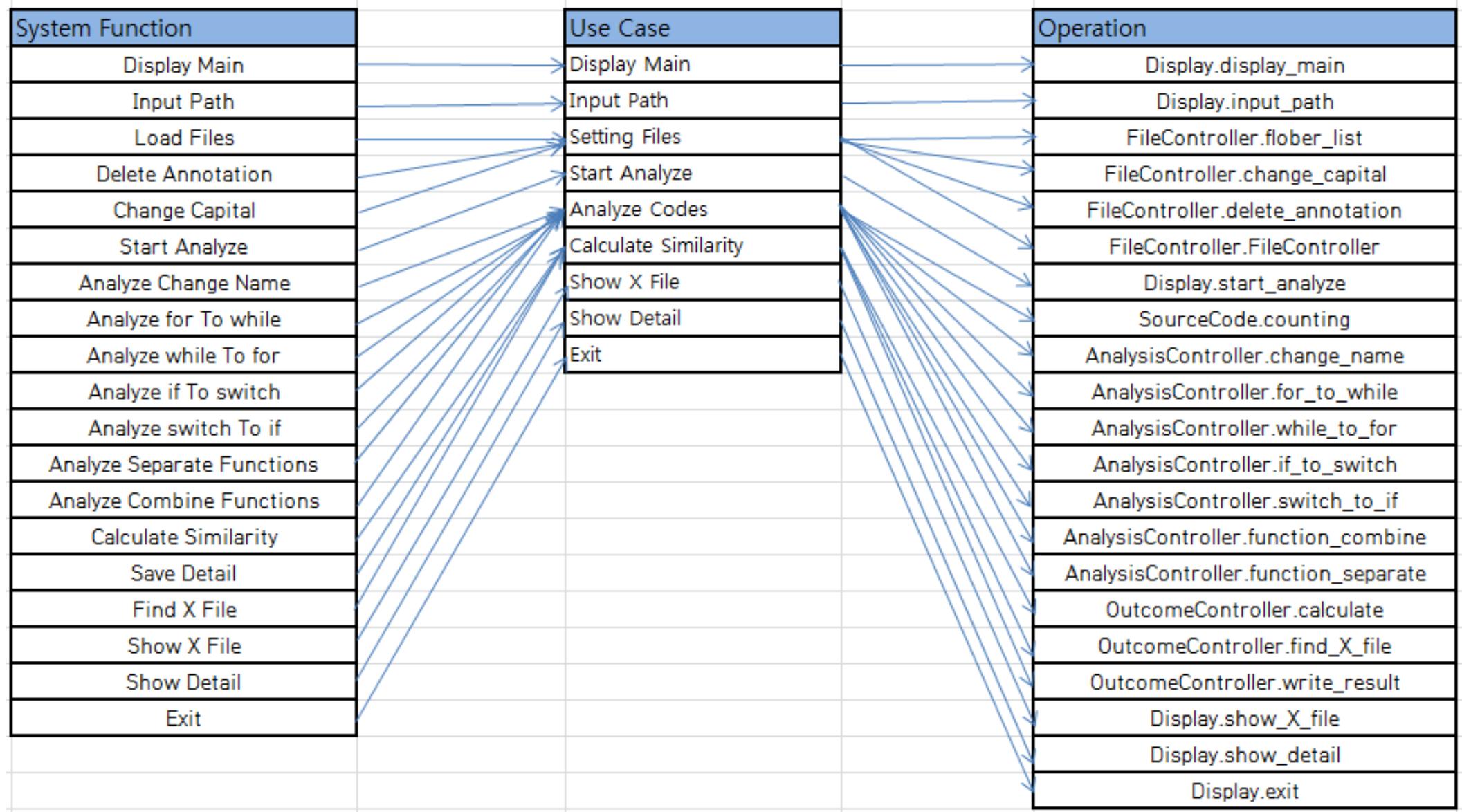
CK.STC.213	switch 문을 if 문으로 바꾼 코드를 잘 찾아 냈는지 검사한다.
CK.STC.214	for 문을 while 문으로 바꾼 코드를 잘 찾아 냈는지 검사한다.
CK.STC.215	while 문을 for 문으로 바꾼 코드를 잘 찾아 냈는지 검사한다.
CK.STC.216	함수를 고의적으로 나눈 코드를 잘 찾아 냈는지 검사한다.
CK.STC.217	함수를 고의적으로 합친 코드를 잘 찾아 냈는지 검사한다.
CK.STC.300	단계별 검사로 나온 결과들을 가지고 유사도를 계산을 잘 하는지 검사한다.
CK.STC.301	분석한 내용들이 파일에 잘 저장되는지 검사한다.
CK.STC.302	유사도를 토대로 코드들 중 가장 중심이 될 것으로 예상되는 파일을 제대로 찾는지 검사한다.
CK.STC.401	유사도 검사 진행을 통해서 코드들 중 가장 중심이 될 것이다 예상되는 파일을 제대로 보여주는지 검사한다.
CK.STC.402	유사도 검사 진행중 분석한 내용을 저장해둔 파일을 제대로 보여주는지 검사한다.
CK.STC.500	유사도 검사가 끝난 뒤에 사용자가 원한대로 종료가 되는 지를 검사한다.



# 8. Analyze Traceability Analysis



# Analyze Traceability Analysis





THANK YOU