

OOPT Stage 2030

<Analysis>

Software Modeling & Analysis

소프트웨어 모델링 및 분석

보고서 Version. 4

Team. T1

201111388 조연호

201211374 이창오

201211379 장종훈

201314196 양동혁

Stage 2030. Analysis

1. Phase 2110. Revise Plan	3
2. Phase 2120. Synchronize Artifacts.....	3
3. Phase 2130. Analyze	3
A. Activity 2131. Define Essential Use Cases	3
B. Activity 2132. Refine Use Case Diagrams.....	14
C. Activity 2133. Define Domain Model	15
D. Activity 2134. Refine Glossary	15
E. Activity 2135. Define System Sequence Diagrams.....	21
F. Activity 2136. Define Operation Contracts	23
G. Activity 2137. Define State Diagrams	26
H. Activity 2138. Refine System Test Case.....	27
I. Activity 2139. Analyze (2030) Traceability Analysis	31

Phase 2110. Revise Plan

OOPT Stage 1000 <Plan and Elaboration> 보고서 Version. 6 수정

Phase 2120. Synchronize Artifacts

OOPT Stage 1000 <Plan and Elaboration> 보고서 Version. 6 수정

Phase 2130. Analyze

Activity 2131. Define Essential Use Cases

Use Case	Select Folder
Actor	User
Purpose	비교할 소스 코드 파일들이 들어있는 폴더를 선택한다.
Overview	사용자가 소스 코드 파일들이 들어있는 폴더 경로를 선택한다.
Type	Primary & Evident
Cross Reference	N/A
Pre-Requisites	버튼(A)를 누른다.
Typical Courses of Events	(A) : Actor, (S) : System 1. (A) : 사용자가 폴더 선택 버튼을 누른다. 2. (S) : 폴더 선택 창을 출력한다. 3. (A) : 사용자가 폴더를 선택한다.
Alternative Courses of Events	N/A
Exceptional Courses of Events	N/A

OOPT Stage 2030 <Analysis>

Use Case	Start
Actor	User
Purpose	지정된 경로의 파일들에 대한 객체들을 생성하고, 비교 결과 화면을 출력한다.
Overview	사용자가 '시작하기' 버튼을 누르면 비교 결과 화면을 출력한다.
Type	Primary & Evident
Cross Reference	R.3(Analyze File)
Pre-Requisites	버튼(B)를 누른다.
Typical Courses of Events	(A) : Actor, (S) : System 1. (A) : 사용자가 '시작하기' 버튼을 누른다. 2. (S) : 비교 결과 화면을 출력한다.
Alternative Courses of Events	N/A
Exceptional Courses of Events	아래의 상황일 경우 폴더 지정 오류 메시지를 출력하고, 결과 화면을 출력하지 않는다. 1. 폴더 안에 .c 확장자를 가진 파일이 없는 경우 2. 폴더 안에 파일이 100개 초과일 경우 3. 폴더 안에 0바이트인 파일이 있는 경우 소스 코드를 읽지 못하는 경우 오류 메시지를 출력하고, 결과 화면을 출력하지 않는다.

OOPT Stage 2030 <Analysis>

Use Case	Analyze File
Actor	-
Purpose	소스 코드 파일을 분석하여 일치율 계산에 필요한 정보로 가공한다.
Overview	시스템 내에서 파일을 분석한다.
Type	Primary & Hidden
Cross Reference	R.3.1(Analyze Line), R.3.2(Analyze Function), R.3.3(Analyze Variable), R.3.4(Analyze Preprocessor), R.3.5(Analyze Annotation)
Pre-Requisites	<i>Files</i> 인스턴스가 생성되어 있어야 한다.
Typical Courses of Events	(A) : Actor, (S) : System 1. (S) : 시스템 내에서 파일을 분석한다.
Alternative Courses of Events	N/A
Exceptional Courses of Events	N/A

Use Case	Change Annotation
Actor	-
Purpose	소스 코드 내의 /* */ 안의 문장을 //문장으로 바꾼다.
Overview	시스템 내에서 주석의 개수를 한 줄씩 셀 수 있도록 바꾼다.
Type	Primary & Hidden
Cross Reference	R.3(Analyze File)
Pre-Requisites	N/A
Typical Courses of Events	(A) : Actor, (S) : System 1. (S) : 소스 코드의 /**/ 주석을 //으로 변경한다.
Alternative Courses of Events	N/A
Exceptional Courses of Events	N/A

OOPT Stage 2030 <Analysis>

Use Case	Analyze Annotation
Actor	-
Purpose	소스 코드 파일의 주석의 개수를 분석하여 결과값을 저장한다.
Overview	시스템 내에서 주석의 개수를 분석한다.
Type	Primary & Hidden
Cross Reference	R.3(Analyze File)
Pre-Requisites	Change Annotation이 실행되어야 한다.
Typical Courses of Events	(A) : Actor, (S) : System 1. (S) : 시스템 내에서 주석의 개수를 분석한다.
Alternative Courses of Events	N/A
Exceptional Courses of Events	N/A

Use Case	Analyze Line
Actor	-
Purpose	소스 코드 파일의 라인 수를 분석하여 결과값을 저장한다.
Overview	시스템 내에서 라인 수를 분석한다.
Type	Primary & Hidden
Cross Reference	R.3(Analyze File)
Pre-Requisites	N/A
Typical Courses of Events	(A) : Actor, (S) : System 1. (S) : 시스템 내에서 라인 수를 분석한다.
Alternative Courses of Events	N/A
Exceptional Courses of Events	N/A

OOPT Stage 2030 <Analysis>

Use Case	Delete Annotation
Actor	-
Purpose	소스 코드 파일의 주석을 삭제한다.
Overview	소스 코드 파일의 주석을 삭제한다.
Type	Primary & Hidden
Cross Reference	R.3(Analyze File)
Pre-Requisites	Analyze Annotation이 실행되어야 한다.
Typical Courses of Events	(A) : Actor, (S) : System 1. (S) : 소스 코드의 주석을 삭제한다.
Alternative Courses of Events	N/A
Exceptional Courses of Events	N/A

Use Case	Divide Code Line
Actor	-
Purpose	한 줄에 같이 작성된 코드들의 행을 나눈다.
Overview	변수나 함수를 ';' 기준으로 분석할 수 있도록 한 줄에 같이 작성된 코드들을 나눈다.
Type	Primary & Hidden
Cross Reference	R.3(Analyze File)
Pre-Requisites	Analyze Line과 Delete Annotation이 실행되어야 한다.
Typical Courses of Events	(A) : Actor, (S) : System 1. (S) : ';' 기준으로 여러 줄로 분할하여 줄을 추가한다.
Alternative Courses of Events	N/A
Exceptional Courses of Events	N/A

OOPT Stage 2030 <Analysis>

Use Case	Delete Printf
Actor	-
Purpose	함수나 변수 분석 결과에 영향을 끼칠 수 있으므로 Printf 문의 괄호 안 내용들을 삭제한다.
Overview	Printf 문의 괄호 안의 내용들을 삭제한다.
Type	Primary & Hidden
Cross Reference	R.3(Analyze File)
Pre-Requisites	Divide Code Line이 실행되어야 한다.
Typical Courses of Events	(A) : Actor, (S) : System 1. (S) : 소스 코드의 Printf문의 괄호 안의 내용들을 삭제한다.
Alternative Courses of Events	N/A
Exceptional Courses of Events	N/A

Use Case	Find Type
Actor	-
Purpose	소스 코드 내 새로 선언된 타입을 찾는다.
Overview	구조체 또는 typedef로 새로 선언된 타입을 찾는다.
Type	Primary & Hidden
Cross Reference	R.3(Analyze File)
Pre-Requisites	Delete Annotation, Divide Code Line, Delete Printf가 실행되어야 한다.
Typical Courses of Events	(A) : Actor, (S) : System 1. (S) : 구조체 또는 typedef로 새로 선언된 타입을 찾는다.
Alternative Courses of Events	N/A
Exceptional Courses of Events	N/A

OOPT Stage 2030 <Analysis>

Use Case	Analyze Function
Actor	-
Purpose	소스 코드 파일의 함수의 개수와 이름을 분석하여 결과값을 저장한다.
Overview	시스템 내에서 함수의 개수와 이름을 분석한다.
Type	Primary & Hidden
Cross Reference	R.3(Analyze File)
Pre-Requisites	Delete Printf, Delete Annotation, Find Type이 실행되어야 한다.
Typical Courses of Events	(A) : Actor, (S) : System 1. (S) : 시스템 내에서 함수의 개수와 이름을 분석한다.
Alternative Courses of Events	N/A
Exceptional Courses of Events	N/A

Use Case	Analyze Variable
Actor	-
Purpose	소스 코드 파일의 변수의 개수와 이름을 분석하여 결과값을 저장한다.
Overview	시스템 내에서 변수의 개수와 이름을 분석한다.
Type	Primary & Hidden
Cross Reference	R.3(Analyze File)
Pre-Requisites	Delete Printf, Delete Annotation, Find Type이 실행되어야 한다.
Typical Courses of Events	(A) : Actor, (S) : System 1. (S) : 시스템 내에서 변수의 개수와 이름을 분석한다.
Alternative Courses of Events	N/A
Exceptional Courses of Events	N/A

OOPT Stage 2030 <Analysis>

Use Case	Analyze Preprocessor
Actor	-
Purpose	소스 코드 파일의 전처리기의 개수와 이름을 분석하여 결과값을 저장한다.
Overview	시스템 내에서 전처리기의 개수와 이름을 분석한다.
Type	Primary & Hidden
Cross Reference	R.3(Analyze File)
Pre-Requisites	Delete Annotation이 실행되어야 한다.
Typical Courses of Events	(A) : Actor, (S) : System 1. (S) : 시스템 내에서 전처리기의 개수와 이름을 분석한다.
Alternative Courses of Events	N/A
Exceptional Courses of Events	N/A

Use Case	Change Center
Actor	User
Purpose	비교 기준이 되는 소스 코드 파일을 변경하여 결과 화면을 갱신한다.
Overview	사용자가 소스 코드 파일을 선택하여 기준을 변경하여 결과 화면을 갱신한다.
Type	Primary & Evident
Cross Reference	R.5(Calculate Sync-Rate)
Pre-Requisites	소스 코드 파일 분석이 정상적으로 완료되어야 한다. (C) 또는 (E)를 누른다.
Typical Courses of Events	(A) : Actor, (S) : System 1. (A) : 사용자가 태그 클라우드에서 소스 코드 파일을 선택한다. 2. (S) : 기준이 변경된다.
Alternative Courses of Events	1. (A) : 사용자가 리스트에서 소스 코드 파일을 선택한다. 2. (S) : 기준이 변경된다.
Exceptional Courses of Events	N/A

OOPT Stage 2030 <Analysis>

Use Case	Calculate File
Actor	-
Purpose	Files 인스턴스에 저장되어있는 변수를 비교하여 일치율을 계산한다.
Overview	Files 인스턴스에 저장되어있는 변수를 바탕으로 시스템 내에서 일치율을 계산한다.
Type	Primary & Hidden
Cross Reference	R.5.1(Calculate Line Sync-Rate), R.5.2(Calculate Function Sync-Rate), R.5.3(Calculate Variable Sync-Rate), R.5.4(Calculate Preprocessor Sync-Rate), R.5.5(Calculate Annotation Sync-Rate)
Pre-Requisites	소스 코드 파일 분석이 정상적으로 완료되어야 한다.
Typical Courses of Events	(A) : Actor, (S) : System 1. (S) : 분석된 데이터를 비교하여 시스템 내에서 일치율을 계산한다.
Alternative Courses of Events	N/A
Exceptional Courses of Events	N/A

Use Case	Calculate Line Sync-Rate
Actor	-
Purpose	분석한 라인 수를 비교하여 일치율을 계산한다.
Overview	분석한 라인 수를 바탕으로 시스템 내에서 일치율을 계산한다.
Type	Primary & Hidden
Cross Reference	R.5(Calculate Sync-Rate)
Pre-Requisites	N/A
Typical Courses of Events	(A) : Actor, (S) : System 1. (S) : 분석한 라인 수를 비교하여 시스템 내에서 일치율을 계산한다.
Alternative Courses of Events	N/A
Exceptional Courses of Events	N/A

OOPT Stage 2030 <Analysis>

Use Case	Calculate Function Sync-Rate
Actor	-
Purpose	분석한 함수의 개수와 이름을 비교하여 일치율을 계산한다.
Overview	분석한 함수의 개수와 이름을 바탕으로 시스템 내에서 일치율을 계산한다.
Type	Primary & Hidden
Cross Reference	R.5(Calculate Sync-Rate)
Pre-Requisites	N/A
Typical Courses of Events	(A) : Actor, (S) : System 1. (S) : 분석한 함수의 개수와 이름을 비교하여 시스템 내에서 일치율을 계산한다.
Alternative Courses of Events	N/A
Exceptional Courses of Events	N/A

Use Case	Calculate Variable Sync-Rate
Actor	-
Purpose	분석한 변수의 개수와 이름을 비교하여 일치율을 계산한다.
Overview	분석한 변수의 개수와 이름을 바탕으로 시스템 내에서 일치율을 계산한다.
Type	Primary & Hidden
Cross Reference	R.5(Calculate Sync-Rate)
Pre-Requisites	N/A
Typical Courses of Events	(A) : Actor, (S) : System 1. (S) : 분석한 변수의 개수와 이름을 비교하여 시스템 내에서 일치율을 계산한다.
Alternative Courses of Events	N/A
Exceptional Courses of Events	N/A

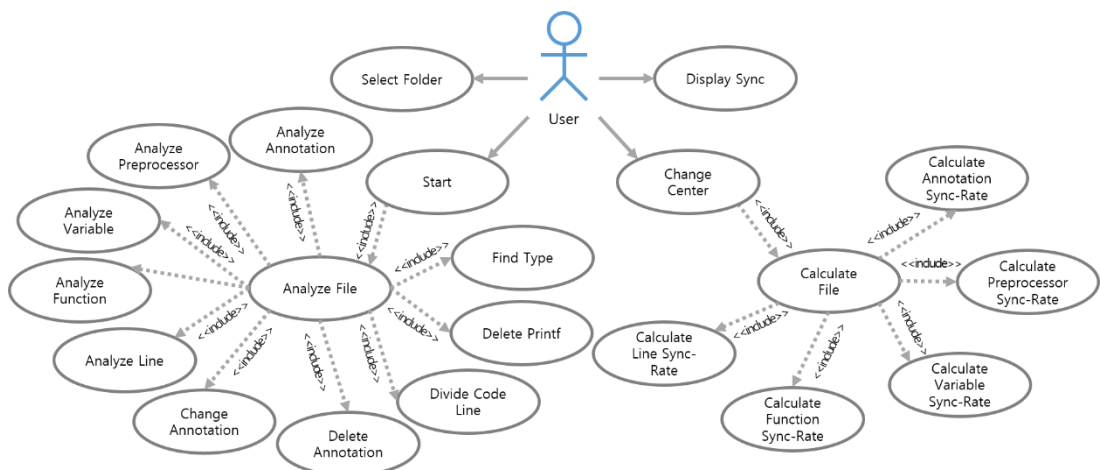
OOPT Stage 2030 <Analysis>

Use Case	Calculate Preprocessor Sync-Rate
Actor	-
Purpose	분석한 전처리기의 개수와 이름을 비교하여 일치율을 계산한다.
Overview	분석한 전처리기의 개수와 이름을 바탕으로 시스템 내에서 일치율을 계산한다.
Type	Primary & Hidden
Cross Reference	R.5(Calculate Sync-Rate)
Pre-Requisites	N/A
Typical Courses of Events	(A) : Actor, (S) : System 1. (S) : 분석한 전처리기의 개수와 이름을 비교하여 시스템 내에서 일치율을 계산한다.
Alternative Courses of Events	N/A
Exceptional Courses of Events	N/A

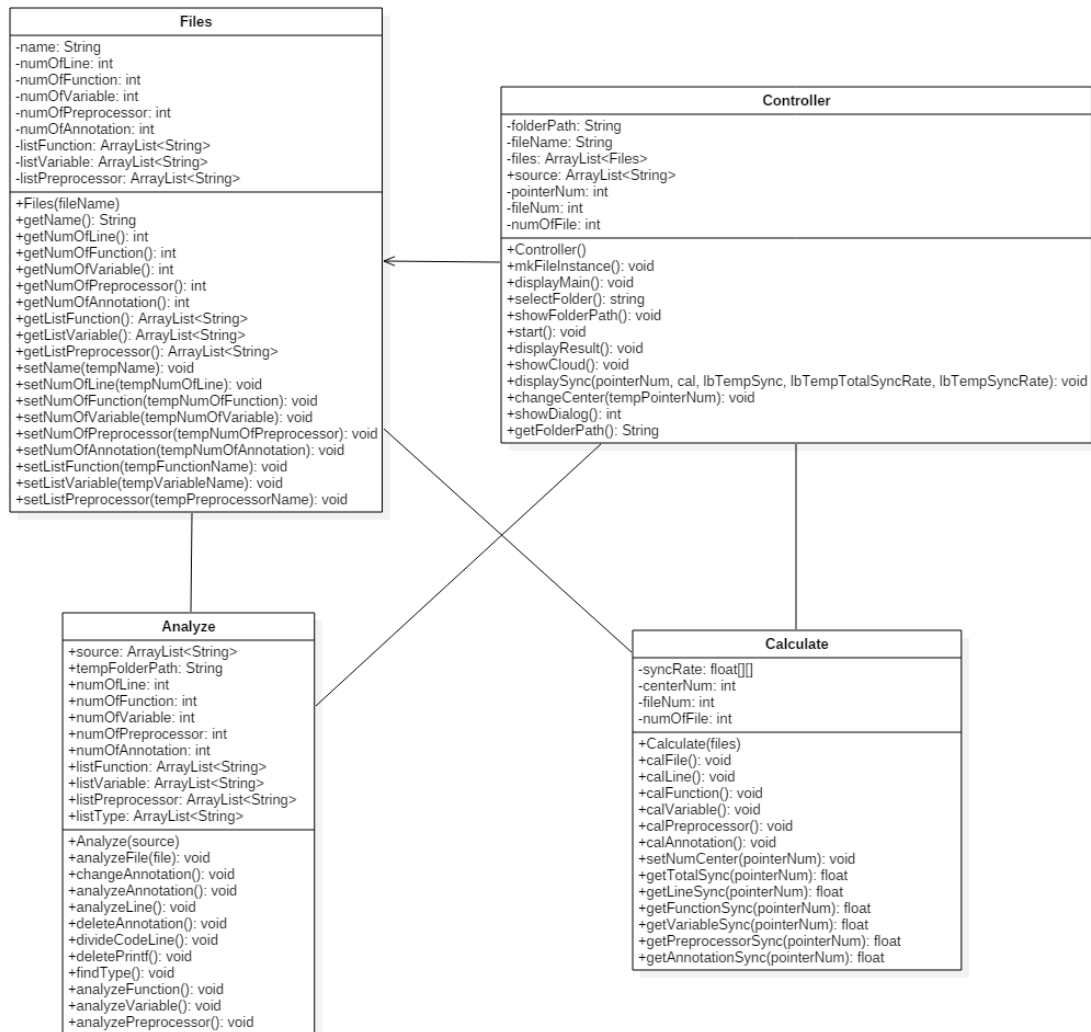
Use Case	Calculate Annotation Sync-Rate
Actor	-
Purpose	분석한 주석의 개수를 비교하여 일치율을 계산한다.
Overview	분석한 주석의 개수를 바탕으로 시스템 내에서 일치율을 계산한다.
Type	Primary & Hidden
Cross Reference	R.5(Calculate Sync-Rate)
Pre-Requisites	N/A
Typical Courses of Events	(A) : Actor, (S) : System 1. (S) : 분석한 주석의 개수를 비교하여 시스템 내에서 일치율을 계산한다.
Alternative Courses of Events	N/A
Exceptional Courses of Events	N/A

Use Case	Display Sync
Actor	User
Purpose	기존 소스 코드 파일과 비교할 소스 코드 파일의 일치율과 유사 항목을 출력한다.
Overview	사용자가 소스 코드 파일 이름에 마우스 커서를 올리면 말풍선 형태로 일치율과 유사 항목을 출력한다.
Type	Primary & Evident
Cross Reference	N/A
Pre-Requisites	소스 코드 파일 비교가 정상적으로 완료되어야 한다.
Typical Courses of Events	(A) : Actor, (S) : System 1. (A) : 사용자가 태그 클라우드의 소스 코드 파일 이름에 마우스 커서를 올린다. 2. (S) : 말풍선 형태로 일치율과 유사 항목을 출력한다.
Alternative Courses of Events	N/A
Exceptional Courses of Events	N/A

Activity 2132. Refine Use Case Diagrams



Activity 2133. Define Domain Model



Activity 2134. Refine Glossary

Term	Category	Remarks
Controller	Class	전반적인 프로그램의 수행을 제어하는 클래스
Analyze	Class	생성된 파일 객체들 안에 저장된 코드를 분석하는 클래스
Calculate	Class	분석된 값을 비교하여 일치율을 계산하는 클래스
Files	Class	각각의 파일들의 소스 코드 및 분석된 값을 저장하는 클래스
Controller.folderPath:	Attribute	파일의 경로를 저장하는 변수
String		
Controller.fileName:	Attribute	파일의 이름을 저장하는 변수

OOPT Stage 2030 <Analysis>

String		
Controller.files: ArrayList<Files>	Attribute	Files 의 객체를 저장하고 있는 Array List
Controller.source: ArrayList<String>	Attribute	소스 코드의 내용을 저장하고 있는 ArrayList
Controller.pointerNum: int	Attribute	사용자가 클릭한 소스 파일의 파일 번호
Controller.fileNum: int	Attribute	폴더에 들어있는 소스 파일의 번호. 반복문에 사용될 변수
Controller.numOfFile: int	Attribute	폴더에 들어있는 총 소스 파일의 수
Analyze.source: ArrayList<String>	Attribute	소스 코드의 내용을 저장하고 있는 ArrayList
Analyze.tempFolderPath : String	Attribute	폴더의 경로를 controller에서 받아와 저장할 변수
Analyze.numOfLine: int	Attribute	소스 코드의 Line 총 개수
Analyze.numOfFunction: int	Attribute	소스 코드 함수의 총 개수
Analyze.numOfVariable: int	Attribute	소스 코드 변수의 총 개수
Analyze.numOfPreprocessor: int	Attribute	소스 코드 전처리기의 총 개수
Analyze.numOfAnnotation: int	Attribute	소스 코드 주석의 총 개수
Analyze.listFunction: ArrayList<String>	Attribute	소스 코드 함수의 이름을 저장하고 있는 ArrayList
Analyze.listVariable: ArrayList<String>	Attribute	소스 코드 변수의 이름을 저장하고 있는 ArrayList
Analyze.listPreprocessor: ArrayList<String>	Attribute	소스 코드 전처리기의 이름을 저장하고 있는 ArrayList
Analyze.listType: ArrayList<String>	Attribute	자료형과 소스 코드 내에서 선언된 타입의 이 름을 저장하고 있는 ArrayList
Calculate.syncRate: float[][]	Attribute	계산된 일치율을 저장할 2차원 배열
Calculate.centerNum: int	Attribute	기준이 되는 파일의 파일 번호를 저장하는 변 수
Calculate.fileNum: int	Attribute	파일의 번호를 저장하는 변수
Calculate.numOfFile: int	Attribute	폴더에 들어있는 총 파일 개수
Files.name: String	Attribute	파일 객체에 폴더에 들어있는 파일명을 저장

OOPT Stage 2030 <Analysis>

		하는 변수
Files.numOfLine: int	Attribute	파일 객체에 소스 코드의 총 Line 수를 저장하는 변수
Files.numOfFunction: int	Attribute	파일 객체에 소스 코드 함수의 개수를 저장하는 변수
Files.numOfVariable: int	Attribute	파일 객체에 소스 코드 변수의 개수를 저장하는 변수
Files.numOfPreprocessor: int	Attribute	파일 객체에 소스 코드 전처리기의 개수를 저장하는 변수
Files.numOfAnnotation: int	Attribute	파일 객체에 소스 코드 주석의 개수를 저장하는 변수
Files.listFunction: ArrayList<String>	Attribute	파일 객체에 소스 코드 함수의 이름을 저장하는 ArrayList
Files.listVariable: ArrayList<String>	Attribute	파일 객체에 소스 코드 변수의 이름을 저장하는 ArrayList
Files.listPreprocessor: ArrayList<String>	Attribute	파일 객체에 소스 코드 전처리기의 이름을 저장하는 ArrayList
Controller.Controller()	Operation	Controller 클래스의 생성자 메소드
Controller.mkFileInstance(): void	Operation	소스 코드 파일들의 객체를 생성하는 메소드
Controller.displayMain(): void	Operation	시작 화면 출력하는 메소드
Controller.selectFolder(): String	Operation	사용자가 폴더를 선택할 수 있도록 하는 메소드
Controller.showFolderPath(): void	Operation	선택된 폴더 경로를 출력하는 메소드
Controller.start(): void	Operation	시작하기 버튼에 대한 동작을 실시하는 메소드
Controller.displayResult(): void	Operation	결과 화면 출력하는 메소드
Controller.showCloud(): void	Operation	클라우드 출력하는 메소드
Controller.displaySync(pointerNum, cal, lbTempSync, lbTempTotalSyncRate, lbTempSyncRate): void	Operation	소스 코드 파일의 일치율을 출력하는 메소드
Controller.changeCenter	Operation	기준 소스 코드 파일을 변경하는 메소드

OOPT Stage 2030 <Analysis>

(tempPointerNum): void		
Controller.showDialog(): int	Operation	폴더 선택 창을 출력하는 메소드
Controller.getFolderPath() (): String	Operation	String 변수 <i>folderPath</i> 를 반환하는 메소드
Analyze.Analyze(source)	Operation	Analyze 클래스의 생성자 메소드
Analyze.analyzeFile(file): void	Operation	전반적인 분석 기능을 수행하고 값을 저장하는 메소드
Analyze.changeAnnotation(): void	Operation	주석 분석 전에 <i>/**/</i> 으로 주석 처리된 줄을 <i>//</i> 으로 변경하는 메소드
Analyze.analyzeAnnotation(): void	Operation	주석 처리된 줄의 개수를 계산하는 메소드
Analyze.analyzeLine(): void	Operation	코드의 라인 수를 계산하는 메소드
Analyze.deleteAnnotation(): void	Operation	분석 기능에 불필요한 주석 처리된 부분을 삭제하는 메소드
Analyze.divideCodeLine(): void	Operation	한 줄 안에 코드가 여러 개인 경우 여러 줄로 분할하는 메소드
Analyze.deletePrintf(): void	Operation	분석 기능에 불필요한 printf문의 괄호 안의 내용들을 삭제하는 메소드
Analyze.findType(): void	Operation	소스 코드 내에서 선언된 타입을 찾는 메소드
Analyze.analyzeFunction(): void	Operation	코드의 함수 수, 이름을 분석하는 메소드
Analyze.analyzeVariable(): void	Operation	코드의 변수 수, 이름을 분석하는 메소드
Analyze.analyzePreprocessor(): void	Operation	코드의 전처리기 수, 이름을 분석하는 메소드
Calculate.Calculate()	Operation	Calculate 클래스의 생성자 메소드
Calculate.calFile(): void	Operation	분석된 값을 바탕으로 전반적인 비교 기능을 수행하는 메소드
Calculate.calLine(): void	Operation	두 파일 간 라인 수를 비교, 일치율을 계산하는 메소드
Calculate.calFunction(): void	Operation	두 파일 간 함수를 비교, 일치율을 계산하는 메소드
Calculate.calVariable(): void	Operation	두 파일 간 변수를 비교, 일치율을 계산하는 메소드
Calculate.calPreprocessor(): void	Operation	두 파일 간 전처리기를 비교, 일치율을 계산하는 메소드

OOPT Stage 2030 <Analysis>

Calculate.calAnnotation()): void	Operation	두 파일 간 주석 수를 비교, 일치율을 계산하는 메소드
Calculate.setNumCenter (pointerNum): void	Operation	선택된 기준 파일의 번호를 저장하는 메소드
Calculate.getTotalSync(p ointerNum): float	Operation	총 일치율을 반환하는 메소드
Calculate.getLineSync(p ointerNum): float	Operation	라인 수 일치율을 반환하는 메소드
Calculate.getFunctionSy nc(pointerNum): float	Operation	함수 일치율을 반환하는 메소드
Calculate.getVariableSyn c(pointerNum): float	Operation	변수 일치율을 반환하는 메소드
Calculate.getPreprocess orSync(pointerNum): float	Operation	전처리기 일치율을 반환하는 메소드
Calculate.getAnnotation Sync(pointerNum): float	Operation	주석 수 일치율을 반환하는 메소드
Files.Files(fileName)	Operation	Files 클래스의 생성자 메소드
Files.getName(): String	Operation	파일 이름을 반환하는 메소드
Files.getNumOfLine(): int	Operation	라인 수를 반환하는 메소드
Files.getNumOfFunction (): int	Operation	함수 수를 반환하는 메소드
Files.getNumOfVariable(): int	Operation	변수 수를 반환하는 메소드
Files.getNumOfPreproce ssor(): int	Operation	전처리기 수를 반환하는 메소드
Files.getNumOfAnnotati on(): int	Operation	주석 수를 반환하는 메소드
Files.getListFunction(): ArrayList<String>	Operation	함수 이름이 저장된 ArrayList를 반환하는 메소드
Files.getListVariable(): ArrayList<String>	Operation	변수 이름이 저장된 ArrayList를 반환하는 메소드
Files.getListPreprocessor (): ArrayList<String>	Operation	전처리기 이름이 저장된 ArrayList를 반환하는 메소드
Files.setName(tempNam e): void	Operation	파일 이름을 저장하는 메소드
Files.setNumOfLine(tem	Operation	라인 수를 저장하는 메소드

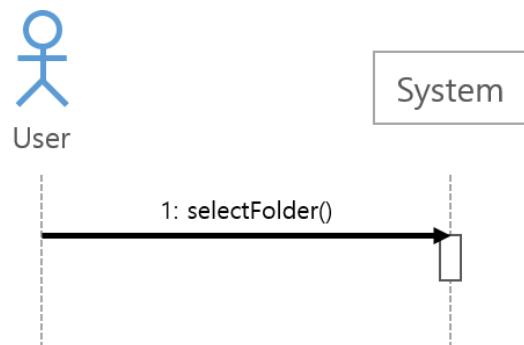
OOPT Stage 2030 <Analysis>

pNumOfLine): void		
Files.setNumOfFunction(tempNumOfFunction): void	Operation	함수 수를 저장하는 메소드
Files.setNumOfVariable(tempNumOfVariable): void	Operation	변수 수를 저장하는 메소드
Files.setNumOfPreprocessor(tempNumOfPreprocessor): void	Operation	전처리기 수를 저장하는 메소드
Files.setNumOfAnnotation(tempNumOfAnnotation): void	Operation	주석 수를 저장하는 메소드
Files.setListFunction(tempFunctionName): void	Operation	함수 이름을 저장하는 메소드
Files.setVariableList(tempVariableName): void	Operation	변수 이름을 저장하는 메소드
Files.setPreprocessorList(tempPreprocessorName): void	Operation	전처리기 이름을 저장하는 메소드

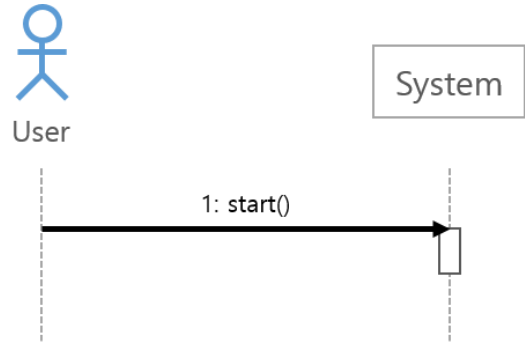
Activity 2135. Define System Sequence Diagrams

Use Case	Name of Actor-Activated Event
1. Select Folder	1: selectFolder()
2. Start	1: start()
3. Analyze File	N/A
4. Change Annotation	N/A
5. Analyze Annotation	N/A
6. Analyze Line	N/A
7. Delete Annotation	N/A
8. Divide Code Line	N/A
9. Delete Printf	N/A
10. Find Type	N/A
11. Analyze Function	N/A
12. Analyze Variable	N/A
13. Analyze Preprocessor	N/A
14. Change Center	1: changeCenter()
15. Calculate File	N/A
16. Calculate Line Sync-Rate	N/A
17. Calculate Function Sync-Rate	N/A
18. Calculate Variable Sync-Rate	N/A
19. Calculate Preprocessor Sync-Rate	N/A
20. Calculate Annotation Sync-Rate	N/A
21. Display Sync	1: displaySync()

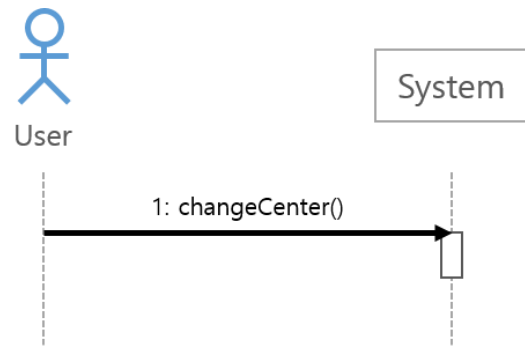
Use Case
Select Folder
<ol style="list-style-type: none"> 1. 사용자가 폴더 선택 버튼을 누른다. 2. 폴더 선택 화면을 출력한다. 3. 사용자가 폴더를 선택한다.



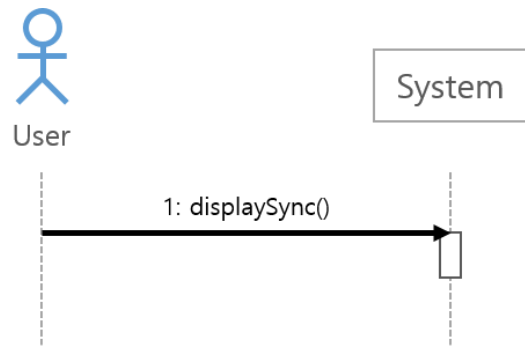
Use Case
Start
<ol style="list-style-type: none"> 1. 사용자가 '시작하기' 버튼을 누른다. 2. 비교 결과 화면을 출력한다.
<ol style="list-style-type: none"> 1. 사용자가 '시작하기' 버튼을 누른다. 2. 예외 발생 시 오류 메시지를 출력한다.



Use Case
Change Center
<ol style="list-style-type: none"> 1. 사용자가 태그 클라우드 또는 리스트에 출력된 소스 코드 파일을 선택한다. 2. 비교 기준이 되는 소스 코드 파일이 변경된다.



Use Case
Display Sync
<ol style="list-style-type: none"> 1. 사용자가 태그 클라우드의 소스 코드 파일 이름에 마우스 커서를 올린다. 2. 말풍선 형태로 일치율과 유사 항목을 출력한다.



Activity 2136. Define Operation Contracts

Use Case	Name of Actor-Activated Event	System Operations
1. Select Folder	1: selectFolder()	selectFolder();
2. Start	1: start()	start()
3. Analyze File	N/A	analyzeFile()
4. Change Annotation	N/A	changeAnnotation()
5. Analyze Annotation	N/A	analyzeAnnotation()
6. Analyze Line	N/A	analyzeLine()
7. Delete Annotation	N/A	deleteAnnotation()
8. Divide Code Line	N/A	divideCodeLine()
9. Delete Printf	N/A	deletePrintf()
10. Find Type	N/A	findType()
11. Analyze Function	N/A	analyzeFunction()
12. Analyze Variable	N/A	analyzeVariable()
13. Analyze Preprocessor	N/A	analyzePreprocessor()
14. Change Center	1: changeCenter()	changeCenter();
15. Calculate File	N/A	calFile();
16. Calculate Line Sync-Rate	N/A	calLine();
17. Calculate Function Sync-Rate	N/A	calFunction();
18. Calculate Variable Sync-Rate	N/A	calVariable();
19. Calculate Preprocessor Sync-Rate	N/A	calPreprocessor();
20. Calculate Annotation Sync-Rate	N/A	calAnnotation();
21. Display Sync	1: displaySync()	displaySync()

Name	selectFolder()
Responsibilities	비교할 소스 코드 파일들이 들어있는 폴더를 선택하는 창을 출력한다.
Type	System
Cross Reference	N/A
Notes	사용자가 소스 코드 파일이 들어있는 폴더를 선택한다.
Exception	비정상적인 폴더 경로일 경우 폴더 선택 창이 닫히지 않는다.
Output	메인 화면 창에 사용자가 선택한 폴더 경로가 나타난다.
Pre-Conditions	사용자가 메인 화면 창에서 폴더 선택 버튼을 누른다.
Post-Conditions	N/A

Name	start()
Responsibilities	지정된 경로의 파일들에 대한 객체들을 생성하고, 비교 결과 화면을 출력한다.
Type	System
Cross Reference	System Function : R.3 Use Case : Analyze File
Notes	사용자가 '시작하기' 버튼을 누른다.
Exception	N/A
Output	비교 결과 화면이 출력된다.
Pre-Conditions	사용자가 폴더 경로를 선택하고, '시작하기' 버튼을 누른다.
Post-Conditions	<i>Controller.folderPath</i> 에 선택한 폴더 경로가 저장된다.

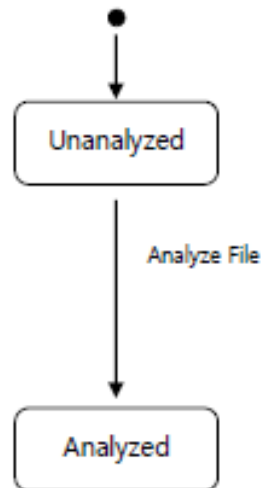
Name	changeCenter()
Responsibilities	비교 기준이 되는 소스 코드 파일을 변경한다.
Type	System
Cross Reference	System Function : R.5 Use Case : Calculate File
Notes	사용자가 선택한 소스 코드 파일을 기준으로 설정한다. 만약 처음 프로그램을 실행하였을 경우 첫 번째 소스 코드 파일을 기준으로 설정한다.
Exception	N/A
Output	결과 화면에 나타나는 비교 결과가 변경된다.
Pre-Conditions	사용자가 태그 클라우드 또는 리스트에서 소스 코드 파일 이름을 누른다.
Post-Conditions	<i>Calculate.centerNum</i> 에 기준이 되는 소스 코드 번호가 저장된다.

OOPT Stage 2030 <Analysis>

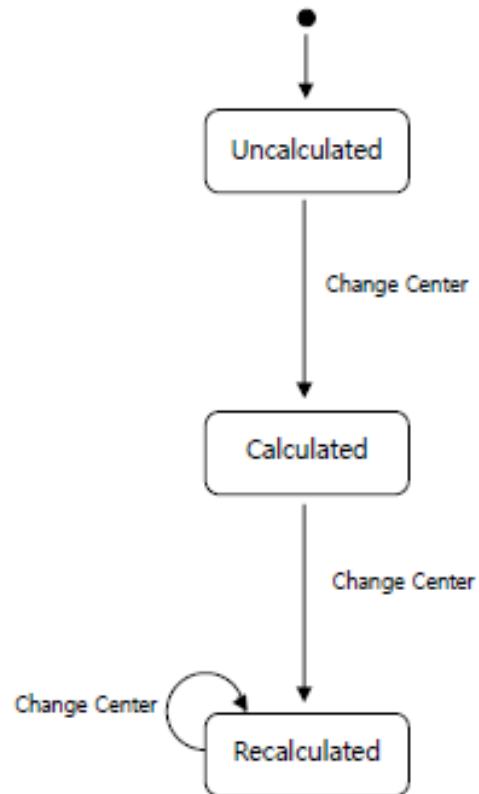
Name	displaySync()
Responsibilities	기준 소스 코드 파일과 비교 소스 코드 파일의 일치율과 유사 항목을 출력한다.
Type	System
Cross Reference	System Function : R.5 Use Case : Calculate File
Notes	사용자가 태그 클라우드의 소스 코드 파일 이름에 마우스 커서를 올리면 일치율과 유사 항목을 출력한다.
Exception	N/A
Output	말풍선 형태로 일치율과 유사 항목을 출력한다.
Pre-Conditions	사용자가 태그 클라우드의 소스 코드 파일 이름에 마우스 커서를 올린다.
Post-Conditions	N/A

Activity 2137. Define State Diagrams

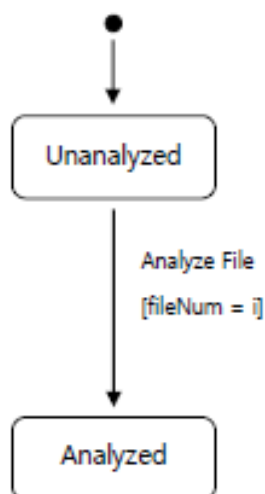
<State Diagram for "Analyze">



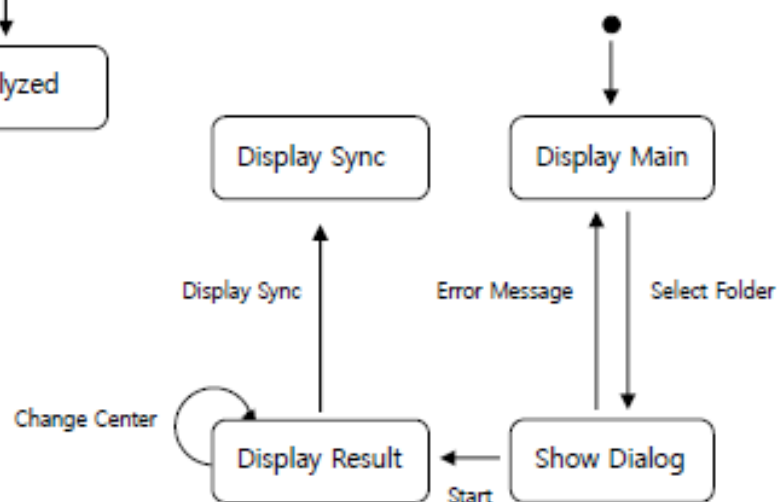
<State Diagram for "Calculate">



<State Diagram for "File">



<State Diagram for "Controller">



Activity 2138. Refine System Test Case

Test Number	Test 항목	Description	Use Case	System Function
1-1	폴더 선택 시험	사용자가 선택한 폴더의 경로가 변수 <i>Controller.folderPath</i> 에 잘 들어갔는지 확인한다.	1. Select Folder	R.1
1-2	폴더 선택 시험	폴더 선택 버튼을 눌렀을 때 폴더 선택 창이 잘 나타나는지 확인한다.	1. Select Folder	R.1
1-3	폴더 선택 시험	사용자가 선택한 폴더의 경로가 정상적으로 출력되는지 확인한다.	1. Select Folder	R.1
1-4	폴더 선택 시험	비정상적인 폴더 경로를 선택하였을 경우 오류 메시지가 팝업 창으로 나타나는지 확인한다.	1. Select Folder	R.1
1-5	폴더 선택 시험	폴더 선택이 완료되었을 경우 '시작하기' 버튼이 활성화되는지 확인한다.	1. Select Folder	R.1
2-1	파일 분석 시작 시험	'시작하기' 버튼을 누르면 파일 분석이 시작되는지 확인한다.	2. Start	R.2
2-2	파일 분석 시작 시험	Files 인스턴스가 실제 소스 코드 파일 개수만큼 생성되는지 확인한다.	2. Start	R.2
2-3	파일 분석 시작 시험	파일 분석이 끝난 후 결과 화면 창이 출력되는지 확인한다.	2. Start	R.2
2-4	파일 분석 시작 시험	.c 확장자가 없는 폴더를 선택했을 경우 오류 메시지가 출력되는지 확인한다.	2. Start	R.2
2-5	파일 분석 시작 시험	파일 100개 초과인 폴더를 선택했을 경우 오류 메시지가 출력되는지 확인한다.	2. Start	R.2
2-6	파일 분석 시작 시험	파일 크기가 0바이트인 파일이 있는 폴더를 선택했을 경우 오류 메시지가 출력되는지 확인한다.	2. Start	R.2
3-1	파일 분석 시험	소스 코드 분석 결과가 올바른지 확인한다.	3. Analyze File	R.3
3-2	파일 분석 시험	소스 코드 분석 시간이 5초 이내인지 확인한다.	3. Analyze File	R.3
4	주석 구조 변경 시험	소스 코드 파일의 주석 형태만 변경하면서 예상한 결과값과 프로그램	4. Change Annotation	R.3.1

OOPT Stage 2030 <Analysis>

		실행 후 얻어지는 결과값을 비교해 본다.		
5	주석 분석 시험	소스 코드 파일의 주석 개수만 변경 하면서 예상한 결과값과 프로그램 실행 후 얻어지는 결과값을 비교해 본다.	5. Analyze Annotation	R.3.2
6	라인 개수 분석 시험	소스 코드 파일의 라인 개수만 변경 하면서 예상한 결과값과 프로그램 실행 후 얻어지는 결과값을 비교해 본다.	6. Analyze Line	R.3.3
7	주석 삭제 시험	소스 코드 파일의 주석 개수만 변경 하면서 예상한 결과값과 프로그램 실행 후 얻어지는 결과값을 비교해 본다.	7. Delete Annotation	R.3.4
8	소스 코드 개행 시험	소스 코드 파일의 한 줄로 같이 작성된 코드의 개수를 변경하면서 예상한 결과값과 프로그램 실행 후 얻어지는 결과값을 비교해 본다.	8. Divide Code Line	R.3.5
9	출력 문자 제외 시험	소스 코드 파일의 'printf()' 메소드로 출력된 코드의 개수를 변경하면서 예상한 결과값과 프로그램 실행 후 얻어지는 결과값을 비교해 본다.	9. Delete Printf	R.3.6
10-1	타입 분석 시험	소스 코드 파일의 typedef나 struct의 개수를 변경하면서 예상한 결과값과 프로그램 실행 후 얻어지는 결과값을 비교해 본다.	10. Find Type	R.3.7
10-2	타입 분석 시험	소스 코드 파일의 typedef나 struct으로 선언된 타입의 이름을 변경하면서 예상한 결과값과 프로그램 실행 후 얻어지는 결과값을 비교해 본다.	10. Find Type	R.3.7
11-1	함수 분석 시험	소스 코드 파일의 함수의 개수를 변경하면서 예상한 결과값과 프로그램 실행 후 얻어지는 결과값을 비교해 본다.	11. Analyze Function	R.3.8
11-2	함수 분석 시험	소스 코드 파일의 함수의 이름을 변경하면서 예상한 결과값과 프로그램 실행 후 얻어지는 결과값을 비교해 본다.	11. Analyze Function	R.3.8

OOPT Stage 2030 <Analysis>

12-1	변수 분석 시험	소스 코드 파일의 변수의 개수를 변경하면서 예상한 결과값과 프로그램 실행 후 얻어지는 결과값을 비교해 본다.	12. Analyze Variable	R.3.9
12-2	변수 분석 시험	소스 코드 파일의 변수의 이름을 변경하면서 예상한 결과값과 프로그램 실행 후 얻어지는 결과값을 비교해 본다.	12. Analyze Variable	R.3.9
13-1	전처리기 분석 시험	소스 코드 파일의 전처리기의 개수를 변경하면서 예상한 결과값과 프로그램 실행 후 얻어지는 결과값을 비교해 본다.	13. Analyze Preprocessor	R.3.10
13-2	전처리기 분석 시험	소스 코드 파일의 전처리기의 이름을 변경하면서 예상한 결과값과 프로그램 실행 후 얻어지는 결과값을 비교해 본다.	13. Analyze Preprocessor	R.3.10
14-1	기준 파일 변경 시험	태그 클라우드 또는 리스트에서 소스 코드 파일 이름을 선택했을 경우 기준이 정상적으로 변경되는지 확인한다.	14. Change Center	R.4
14-2	기준 파일 변경 시험	기준이 변경되면서 비교 결과가 정상적으로 변경되는지 확인한다.	14. Change Center	R.4
14-3	기준 파일 변경 시험	현재 기준인 소스 코드 파일 이름을 다시 선택했을 경우 비교가 다시 되는 않는지 확인한다.	14. Change Center	R.4
14-4	기준 파일 변경 시험	기준 소스 코드 파일이 2초 이내에 변경되는지 확인한다.	14. Change Center	R.4
15-1	일치율 계산 시험	계산 결과가 올바른지 확인한다.	15. Calculate File	R.5
15-2	일치율 계산 시험	계산이 8초 이내에 이루어지는지 확인한다.	15. Calculate File	R.5
16	라인 개수 일치율 계산 시험	소스 코드 파일의 라인 개수만 변경하면서 예상한 결과값과 프로그램 실행 후 얻어지는 결과값을 비교해 본다.	16. Calculate Line Sync-Rate	R.5.1
17	함수 일치율 계산 시험	소스 코드 파일의 함수의 개수 및 이름 등을 변경하면서 예상한 결과값과 프로그램 실행 후 얻어지는 결	17. Calculate Function Sync-Rate	R.5.2

OOPT Stage 2030 <Analysis>

		과값을 비교해 본다.		
18	변수 일치율 계산 시험	소스 코드 파일의 변수의 개수 및 이름 등을 변경하면서 예상한 결과값과 프로그램 실행 후 얻어지는 결과값을 비교해 본다.	18. Calculate Variable Sync-Rate	R.5.3
19	전처리기 일치율 계산 시험	소스 코드 파일의 전처리기의 개수 및 이름 등을 변경하면서 예상한 결과값과 프로그램 실행 후 얻어지는 결과값을 비교해 본다.	19. Calculate Preprocessor Sync-Rate	R.5.4
20	주석 일치율 계산 시험	소스 코드 파일의 주석의 개수 및 이름 등을 변경하면서 예상한 결과값과 프로그램 실행 후 얻어지는 결과값을 비교해 본다.	20. Calculate Annotation Sync-Rate	R.5.5
21-1	일치율 출력 시험	마우스 커서가 올라간 소스 코드 파일의 비교율이 정상적으로 출력되는지 확인한다.	21. Display Sync	R.6
21-2	일치율 출력 시험	다른 오브젝트와 겹쳐서 출력되는 부분이 없는지 확인한다.	21. Display Sync	R.6

Activity 2139. Analyze (2030) Traceability Analysis

