System Testing Respond Report

Software Modeling & Analysis

소프트웨어 모델링 및 분석

보고서 Version. 1

Team. T1

201111388 조연호

201211374 이창오

201211379 장종훈

201314196 양동혁

System Testing Respond Report

1. 1 조 System Test 대응 3
   A. Specification 3
   B. Brute Force Testing 12
   C. Category Partition Testing 15
2. 5 조 System Test 대응 16
   A. Specification 16
   B. Brute Force Testing 20

1. 1조 System Test 대응

1. Specification

위치	Stage 1000 - Activity 1001. Define Draft Plan
문제	기본 문서에서 프로그램의 Object로 쉽게 보여주는 Clone Checker를 지향하였지만 쉽게 보이지 않았다.
대응	결과 화면 창에 출력되는 클라우드 내 텍스트와 리스트 내 텍스트의 출력 형태와 크기를 변경하였다.

위치	Stage 1000 - Activity 1001. Define Draft Plan
문제	Activity 1001(Define Draft Plan)에서는 Functional Requirements가 4개만 정의되어 있으나 Activity 1003(Define Requirements)에서는 20개로 늘어나 서로 Matching이 되지 않는다.
대응	Activity 1003. Define Requirements에 작성된 내용을 토대로 동기화하였다.

위치	Stage 1000 - Activity 1003. Define Requirements
문제	Activity 1003(Define Requirements)의 Functional Requirements 3.7과 3.8의 설명이 같고 그 설명마저도 모호하다.
대응	R.3.7(Analyze Function)의 설명을 '소스 코드 파일의 함수의 개수와 이름을 분석하여 결과값을 저장한다.'로 수정하였다. R.3.8(Analyze Variable)의 설명을 '소스 코드 파일의 변수의 개수와 이름을 분석하여 결과값을 저장한다.'로 수정하였다.

위치	Stage 1000 - Activity 1009. Define System Test Case
문제	Activity 1009(Define System Test Case)에서 Functional Requirements Test Case의 R3.1과 R3.2 의 설명이 같다.
대응	R.3.1(Change Annotation)의 설명을 '소스 코드 파일의 주석 형태만 변경하면서 예상한 결과값과 프로그램 실행 후 얻어지는 결과값을 비교해 본다.'로 수정하였다. R.3.2(Analyze Annotation)의 설명을 '소스 코드 파일의 주석을 변경하면서 예상한 결과값과 프로그램 실행 후 얻어지는 결과값을 비교해 본다.'로 수정하였다.

System Testing Respond Report

위치	Stage 2030 - Activity 2131. Define Essential Use Cases Stage 1000 - Activity 1007. Define Business Use Case
문제	OSP 2030 Stage의 Activity 2131(Define Essential Use Cases)에서 Start와 관련된 Use Case와 OSP 1000 Stage의 Activity 1007(Define Business Use Case)에서 Describe Use Cases의 Start use case의 설명이 서로 다르다.
대응	Stage 1000 - Activity 1007. Define Business Use Case의 Description과 Stage 2030 - Activity 2131. Define Essential Use Cases의 Purpose의 내용을 '지정된 경로의 파일들에 대한 객체들을 생성하고, 비교 결과 화면을 출력한다.'로 동기화하였다.

위치	Stage 2030 - Activity 2131. Define Essential Use Cases Stage 1000 - Activity 1007. Define Business Use Case
문제	OSP 2030 Stage의 Activity 2131(Define Essential Use Cases)에서 Delete Annotation에 관련된 Use Case와 OSP 1000 Stage의 Activity 1007(Define Business Use Case)에서 Describe Use Cases)의 Delete Annotation use case의 설명이 서로 다르다.
대응	Stage 1000 - Activity 1007. Define Business Use Case의 Description과 Stage 2030 - Activity 2131. Define Essential Use Cases의 Purpose의 내용을 '소스 코드 파일의 주석을 삭제한다.'로 동기화하였다.

위치	Stage 2030 - Activity 2131. Define Essential Use Cases Stage 1000 - Activity 1007. Define Business Use Case
문제	OSP 2030 Stage의 Activity 2131(Define Essential Use Cases)에서 Divide Code Line와 관련된 Use Case의 코드 분할기준이 OSP 1000 Stage의Activity 1007(Define Business Use Case)의 Describe Use Cases에서 Divide Code Line use case의 설명에 비해 다소 추상적이다.
대응	Stage 1000 - Activity 1007. Define Business Use Case의 Description과 Stage 2030 - Activity 2131. Define Essential Use Cases의 Purpose의 내용을 '소스 코드 파일에서 한 줄에 여러 개의 코드가 있는 경우 ';'을 기준으로 행을 나눈다.'로 구체화하여 동기화하였다.

위치	Stage 2030 - Activity 2131. Define Essential Use Cases Stage 2040 - Activity 2141. Design Real Use Cases
문제	OSP 2030 Stage의 Activity 2131(Define Essential Use Cases)에서 Divide Code Line에서 있었던 Exceptional Courses of Events가 OSP 2040의 Divide Code Line Use Case에서 없어졌다.
대응	두 곳 모두 Exceptional Courses of Events를 제거하였다.

System Testing Respond Report

위치	Stage 2040 - Activity 2141. Design Real Use Cases
문제	OSP 2040 Stage의 Use Case "Select folder"에서 Pre Requisites의 내용과 Typical Courses of Events의 내용이 겹치고 Exception처리가 반영되어 있어야 하지만 문서상에서는 발견되지 않았다.
대응	Typical Courses of Events에서 Pre Requisites와 중복되었던 '버튼(A)를 누른다.' 내용을 삭제하였다. Exceptional Courses of Events를 제거하였다.

위치	Stage 2040 - Activity 2141. Design Real Use Cases
문제	OSP 2040 Stage의 Use Case "Start"에서 Pre Requisites의 내용과 Typical Courses of Events의 내용이 겹치고 Exception처리가 반영되어 있어야하지만 문서상에서는 발견되지 않았다.
대응	Typical Courses of Events에서 Pre-Requisites와 중복되었던 '버튼(B)를 누른다.' 내용을 삭제하였다. Exceptional Courses of Events의 내용은 아래와 같이 추가하였다. 아래의 상황일 경우 폴더 지정 오류 메시지를 출력하고, 결과 화면을 출력하지 않는다. 1. 폴더 안에 .c 확장자를 가진 파일이 없는 경우 – ".c 확장자를 가진 파일이 없는 폴더입니다." 메시지 2. 폴더 안에 파일이 100개 초과일 경우 – "파일이 100개 이상인 폴더입니다." 메시지 3. 폴더 안에 0바이트인 파일이 있는 경우 – "0Byte 파일이 있는 폴더입니다." 메시지 소스 코드를 읽지 못하는 경우 "폴더 경로가 올바르지 않습니다." 오류 메시지를 출력하고, 결과 화면을 출력하지 않는다.

위치	Stage 2040 - Activity 2141. Design Real Use Cases
문제	OSP 2040 Stage의 Use Case "Analyze File"에서 Pre Requisites의 내용 중 "정상적인 폴더"의 의미가 모호하고 Typical Courses of Events의 순서 중 deletePrintf를 가장 아래에 넣어야 한다.
대응	Purpose를 '함수나 변수 분석 결과에 영향을 끼칠 수 있으므로 Printf 문의 괄호 안 내용들을 삭제한다.'로 설명을 구체화하였다. deletfePrintf()의 위치는 분석 알고리즘 상 문제가 없다고 판단하여 수정하지 않았다.

위치	Stage 2040 - Activity 2141. Design Real Use Cases
문제	OSP 2040 Stage의 Use Case "Change Annotation"에서 Overview의 내용 중 "쉽게"라는 의미가 다소 모호하고 Pre Requisites에 changeAnnotation이 실행되어야 한다가 빠져있고 Typical Courses of Events 의 정의가 애매모호하다.
대응	Overview를 '시스템 내에서 주석의 개수를 한 줄씩 셀 수 있도록 바꾼다.'로 설명을 구체화하였다. Pre-Requisites는 해당 내용이 필요 없으므로 추가하지 않았다. Typical Courses of Events의 내용을 아래와 같이 구체화하였다. (A) : Actor, (S) : System 1. (S) : 생성된 *Analyze:source* 안에 '/* */'로 작성된 주석을 찾는다. 2. (S) : '/*'을 '//'로 바꾸고 '/* */' 주석 안의 문장들 앞에 '//'를 붙인다.

위치	Stage 2040 - Activity 2141. Design Real Use Cases
문제	OSP 2040 Stage의 Use Case "Analyze Annotation"에서 Typical Courses of Events의 정의가 애매모호하다.
대응	Typical Courses of Events의 내용을 아래와 같이 구체화하였다. (A) : Actor, (S) : System 1. (S) : 생성된 *Analyze:source*를 확인한다. 2. (S) : '//' 시작되는 소스 코드 발견 시 주석의 개수 (*File:numOfAnnotation*)를 증가시킨다.

위치	Stage 2040 - Activity 2141. Design Real Use Cases
문제	OSP 2040 Stage의 Use Case "Delete Annotation"의 Pre Requisites에 analyzeAnnotation이 실행되어야 한다가 빠져있고 Typical Courses of Events의 정의가 애매모호하다.
대응	Pre-Requisites에 'Analyze Annotation이 실행되어야 한다.'를 추가하였다. Typical Courses of Events의 내용을 아래와 같이 구체화하였다. (A) : Actor, (S) : System 1. (S) : 생성된 *Analyze:source*를 확인한다. 2. (S) : *Analyze:source*에 '//'을 포함한 문장을 삭제한다.

System Testing Respond Report

위치	Stage 2040 - Activity 2141. Design Real Use Cases
문제	OSP 2040 Stage의 Use Case "Analyze Line"에서 Typical Courses of Events의 1번에서 개수 파악이 어느 개수인지 모르겠다.
대응	Typical Courses of Events의 내용을 아래와 같이 구체화하였다. (A) : Actor, (S) : System 1. (S) : 생성된 *Analyze:source*의 문장 개수를 파악하여 소스 코드 파일의 라인 수를 분석한다. 2. (S) : 라인 수를 *File:numOfLine*에 저장한다.

위치	Stage 2040 - Activity 2141. Design Real Use Cases
문제	OSP 2040 Stage의 Use Case "Divide Code Line"에서 Overview의 내용 중 "쉽게"라는 의미가 다소 모호하고 Pre Requisites에 analyzeLine과 deleteAnnotation이 실행되어야 한다가 빠져있고 Typical Courses of Events의 정의가 애매모호하다.
대응	Overview를 '변수나 함수를 ';' 기준으로 분석할 수 있도록 한 줄에 같이 작성된 코드들을 나눈다.'로 구체화하였다. Pre-Requisites에 'Analyze Line과 Delete Annotation이 실행되어야 한다.'를 추가하였다. Typical Courses of Events의 내용을 아래와 같이 구체화하였다. (A) : Actor, (S) : System 1. (S) : 생성된 *Analyze:source*를 확인한다. 2. (S) : ';'을 기준으로 문자열을 나눈 후 뒤의 내용을 새로운 Array로 만들어 *Analyze:source*에 추가한다.

위치	Stage 2040 - Activity 2141. Design Real Use Cases
문제	OSP 2040 Stage의 Use Case "Delete Printf"에서 목적과 Overview의 내용이 정확하게 작성되지 않았고 Pre Requisites에 Analyze의 모든 상황이 실행된 뒤에 실행되어야 한다가 빠져있고 Typical Courses of Events의 정의가 애매모호하다.
대응	Purpose를 '함수나 변수 분석 결과에 영향을 끼칠 수 있으므로 Printf 문의 괄호 안 내용들을 삭제한다.'로 구체화하였다. Overview를 'Printf 문의 괄호 안의 내용들을 삭제한다.'로 구체화하였다. Pre-Requisites에 'Divide Code Line이 실행되어야 한다.'를 추가하였다. Typical Courses of Events의 내용을 아래와 같이 구체화하였다. (A) : Actor, (S) : System 1. (S) : 생성된 *Analyze:source*를 확인한다. 2. (S) : printf 문의 괄호 안 내용을 삭제한다.

위치	Stage 2040 - Activity 2141. Design Real Use Cases
문제	OSP 2040 Stage의 Use Case "Analyze Function"의 Pre Requisites에 deleteAnnotation이 실행되어야 한다가 빠져있고 Typical Courses of Events의 정의가 애매모호하다.
대응	Pre-Requisites에 'Delete Printf, Delete Annotation, Find Type이 실행되어야 한다.'를 추가하였다. Typical Courses of Events의 내용을 아래와 같이 구체화하였다. (A) : Actor, (S) : System 1. (S) : 생성된 *Analyze:source*를 확인한다. 2. (S) : 리턴 타입(*Analyze:listType*) 발견 시 문장 내에 '(' 가 들어가고 ';'가 들어있지 않다면 함수의 개수(*File:numOfFunction*)를 증가시킨다. 3. (S) : *File:listFunction*에 함수의 이름을 저장한다.

System Testing Respond Report

위치	Stage 2040 - Activity 2141. Design Real Use Cases
문제	OSP 2040 Stage의 Use Case "Analyze Variable"의 Pre Requisites에 deleteAnnotation이 실행되어야 한다가 빠져있고 Typical Courses of Events의 정의가 애매모호하다.
대응	Pre-Requisites에 'Delete Printf, Delete Annotation, Find Type이 실행되어야 한다.'를 추가하였다. Typical Courses of Events의 내용을 아래와 같이 구체화하였다. (A) : Actor, (S) : System 1. (S) : 생성된 *Analyze:source*를 확인한다. 2. (S) : 자료형(*Analyze:listType*) 발견 시 문장 내에 '('가 들어가지 않은 경우, ' , '가 없을 때까지 변수의 개수(*File:numOfVariable*)를 증가시킨다. 3. (S) : *File:listVariable*에 변수의 이름을 저장한다.

위치	Stage 2040 - Activity 2141. Design Real Use Cases
문제	OSP 2040 Stage의 Use Case "Analyze Preprocessor"의 Pre Requisites에 deleteAnnotation이 실행되어야 한다가 빠져있고 Typical Courses of Events의 2번에서 3개의 전처리기를 정의해주었는데 정의해놓은 전처리기 이외에도 다른 전처리기들이 있다.
대응	Pre-Requisites에 'Delete Annotation이 실행되어야 한다.'를 추가하였다. Typical Courses of Events의 내용을 아래와 같이 구체화하였다. (A) : Actor, (S) : System 1. (S) : 생성된 *Analyze:source*를 확인한다. 2. (S) : '#'로 시작되는 문장 발견 시 전처리기의 개수 (*File:numOfPreprocessor*)를 증가시킨다. 3. (S) : *File:listPreprocessor*에 전처리기의 이름을 저장한다.

위치	Stage 2040 - Activity 2141. Design Real Use Cases
문제	OSP 2040 Stage의 Use Case "Change Center"에서 기준점을 변경 후에 결과물을 보여준다는 말이 없다.
대응	Typical Courses of Events의 마지막에 '(S) : 결과 화면을 갱신한다.'를 추가하였다.

<table>
<tr><td>위치</td><td>Stage 2040 - Activity 2141. Design Real Use Cases</td></tr>
<tr><td>문제</td><td>OSP 2040 Stage의 Use Case “Calculate Line Sync-Rate”, “Calculate Function Sync-Rate”, “Calculate Variable Sync-Rate”, “Calculate Preprocessor Sync-Rate”, “Calculate Annotation Sync-Rate”에서 Typical Courses of Events의 내용 중 “오차범위(30,10)”의 내용이 정확히 무엇을 의미하는지 문서의 내용을 읽어만 봐서는 알기 힘들다.</td></tr>
<tr><td>대응</td><td>Calculate Line Sync-Rate의 Typical Courses of Events을 아래와 같이 구체화하였다.
<table>
<tr><td>Typical Courses of Events</td><td>(A) : Actor, (S) : System
1. (S) : 기준 파일의 라인 수가 비교 파일의 라인 수보다 클 경우 ‘(비교 파일의 라인 수 / 기준 파일의 라인 수) * 100’ 수식의 결과값을 일치율로 부여한다.
2. (S) : 일치율을 Calculate:syncRate에 저장한다.</td></tr>
<tr><td>Alternative Courses of Events</td><td>(A) : Actor, (S) : System
1. (S) : 기준 파일의 라인 수가 비교 파일의 라인 수보다 작을 경우 ‘(기준 파일의 라인 수 / 비교 파일의 라인 수) * 100’ 수식의 결과값을 일치율로 부여한다.
2. (S) : 일치율을 Calculate:syncRate에 저장한다.</td></tr>
</table>
Calculate Function Sync-Rate의 Typical Courses of Events을 아래와 같이 구체화하였다.
<table>
<tr><td>Typical Courses of Events</td><td>(A) : Actor, (S) : System
1. (S) : 기준 파일의 함수 수가 비교 파일의 함수 수보다 클 경우 ‘{(비교 파일과 기준 파일에서 이름이 같은 함수 수 / 기준 파일의 함수 수) * 100} * 0.6 + {(비교 파일의 함수 수 / 기준 파일의 함수 수) * 100} * 0.4’ 수식의 결과값을 일치율로 부여한다.
2. (S) : 일치율을 Calculate:syncRate에 저장한다.</td></tr>
<tr><td>Alternative Courses of Events</td><td>(A) : Actor, (S) : System
1. (S) : 기준 파일의 함수 수가 비교 파일의 함수 수보다 작을 경우 ‘{(비교 파일과 기준 파일에서 이름이 같은 함수 수 / 비교 파일의 함수 수) * 100} * 0.6 + {(기준 파일의 함수 수 / 비교 파일의 함수 수) * 100} * 0.4’ 수식의 결과값을 일치율로 부여한다.
2. (S) : 일치율을 Calculate:syncRate에 저장한다.</td></tr>
</table>
Calculate Variable Sync-Rate의 Typical Courses of Events을 아래와 같이 구체화하였다.
<table>
<tr><td>Typical Courses of Events</td><td>(A) : Actor, (S) : System
1. (S) : 기준 파일의 변수 수가 비교 파일의 변수 수보다 클 경우 ‘{(비교 파일과 기준 파일에서 이름이 같은 변수</td></tr>
</table></td></tr>
</table>

	수 / 기준 파일의 변수 수) * 100} * 0.6 + {(비교 파일의 변수 수 / 기준 파일의 변수 수) * 100} * 0.4′ 수식의 결과값을 일치율로 부여한다. 2. (S) : 일치율을 *Calculate:syncRate*에 저장한다.
Alternative Courses of Events	(A) : Actor, (S) : System 1. (S) : 기준 파일의 변수 수가 비교 파일의 변수 수보다 작을 경우 '{(비교 파일과 기준 파일에서 이름이 같은 변수 수 / 비교 파일의 변수 수) * 100} * 0.6 + {(기준 파일의 변수 수 / 비교 파일의 변수 수) * 100} * 0.4′ 수식의 결과값을 일치율로 부여한다. 2. (S) : 일치율을 Calculate:syncRate에 저장한다.

Calculate Preprocessor Sync-Rate의 Typical Courses of Events을 아래와 같이 구체화하였다.

Typical Courses of Events	(A) : Actor, (S) : System 1. (S) : 기준 파일의 전처리기 수가 비교 파일의 전처리기 수보다 클 경우 '{(비교 파일과 기준 파일에서 이름이 같은 전처리기 수 / 기준 파일의 전처리기 수) * 100} * 0.6 + {(비교 파일의 전처리기 수 / 기준 파일의 전처리기 수) * 100} * 0.4′ 수식의 결과값을 일치율로 부여한다. 2. (S) : 일치율을 *Calculate:syncRate*에 저장한다.
Alternative Courses of Events	(A) : Actor, (S) : System 1. (S) : 기준 파일의 전처리기 수가 비교 파일의 전처리기 수보다 작을 경우 '{(비교 파일과 기준 파일에서 이름이 같은 전처리기 수 / 비교 파일의 전처리기 수) * 100} * 0.6 + {(기준 파일의 전처리기 수 / 비교 파일의 전처리기 수) * 100} * 0.4′ 수식의 결과값을 일치율로 부여한다. 2. (S) : 일치율을 Calculate:syncRate에 저장한다.

Calculate Annotation Sync-Rate의 Typical Courses of Events을 아래와 같이 구체화하였다.

Typical Courses of Events	(A) : Actor, (S) : System 1. (S) : 기준 파일의 주석 수가 비교 파일의 주석 수보다 클 경우 '(비교 파일의 주석 수 / 기준 파일의 주석 수) * 100′ 수식의 결과값을 일치율로 부여한다. 2. (S) : 일치율을 *Calculate:syncRate*에 저장한다.
Alternative Courses of Events	(A) : Actor, (S) : System 1. (S) : 기준 파일의 주석 수가 비교 파일의 주석 수보다 작을 경우 '(기준 파일의 주석 수 / 비교 파일의 주석 수) * 100′ 수식의 결과값을 일치율로 부여한다. 2. (S) : 일치율을 Calculate:syncRate에 저장한다.

System Testing Respond Report

위치	Stage 2040 - Activity 2141. Design Real Use Cases
문제	OSP 2040 Stage의 Use Case "Calculate File"의 Overview에서는 분석된 데이터를 바탕으로 시스템이 돌아간다고 되어 있지만 Typical Courses of Events에서는 분석된 데이터의 정의가 있지 않다.
대응	Purpose를 'Files 인스턴스에 저장되어있는 변수를 비교하여 일치율을 계산한다.'로 구체화하였다. Typical Courses of Events을 아래와 같이 구체화하였다. (A) : Actor, (S) : System 1. (S) : 호출 시 전달받은 *Controller.files*를 *Calculate.files*에 저장한다. 2. (S) : *Calculate:syncRate*를 초기화한다 3. (S) : *Calculate:calLine()*을 호출한다. 4. (S) : *Calculate:calFunction()*을 호출한다. 5. (S) : *Calculate:calVariable()*을 호출한다. 6. (S) : *Calculate:calPreprocessor()*을 호출한다. 7. (S) : *Calculate:calAnnotation()*을 호출한다.

2. Brute Force Testing

문제	ST-5. 개당 대용량 소스파일 비교 - Fail ST-6. 소스 파일 분석 - Fail ST-7. 파일 1개 분석 - Fail ST-8. 파일 2개 분석 - Fail ST-9. 파일 3개 분석 - Fail ST-10. 파일 4개 분석 - Fail ST-11. 파일 5개 분석 - Fail ST-12. 파일 6개~10개 분석 - Fail ST-13. 파일 11개~50개 분석 - Fail
원인	한 줄에 같이 작성된 코드들의 행을 나누는 divideCodeLine() 메소드 내에서 무한 루프가 발생되었다.
대응	divideCodeLine() 메소드의 알고리즘을 수정하였다.

문제	ST-14. 오른쪽 버튼 작동 - Fail
원인	결과 화면 창에서 출력되는 클라우드 및 리스트 내 내용이 마우스 왼쪽 클릭뿐만 아니라 오른쪽 클릭과 휠 클릭으로도 동일하게 동작되었다.
대응	마우스 왼쪽 클릭으로만 프로그램을 동작할 수 있도록 수정하였다.

문제	ST-15. 기준점 변경 - Fail
원인	이미 기준 소스 코드 파일로 선택되었던 소스 코드 파일 이름 선택 시 클라우드 모양만 갱신되었다.
대응	현재 기준 소스 코드 파일로 선택되었던 소스 코드 파일 이름을 선택할 경우 기준점이 변경되는 작업을 수행하지 않도록 수정하였다.

문제	ST-16. 폴더만 있었을 때 비교 - Fail ST-23. 소스파일이 없는 빈 폴더 - Fail
원인	사용자가 지정한 폴더에 파일이 없는 경우에도 분석이 시작되는 오류가 발생하였다.
대응	폴더 내 파일이 없는 경우 예외처리를 통해 오류 메시지를 출력하고 다시 시작 화면으로 돌아가도록 수정하였다.

문제	ST-17. 0바이트 파일 비교 - Fail
원인	사용자가 지정한 폴더에 있는 0바이트인 파일을 분석하는 오류가 발생하였다.
대응	폴더 내 0바이트 파일이 있는 경우 예외처리를 통해 오류 메시지를 출력하고 다시 시작 화면으로 돌아가도록 수정하였다.

문제	ST-18. 숨김 파일 비교 - Fail
원인	사용자가 지정한 폴더에 있는 숨김 파일을 분석하는 오류가 발생하였다.
대응	폴더 내 숨김 파일이 있는 경우 예외처리를 통해 오류 메시지를 출력하고 다시 시작 화면으로 돌아가도록 수정하였다.

문제	ST-20. 같은 기준 선택시 변화 - Fail
원인	이미 기준 소스 코드 파일로 선택되어있는 파일로 기준 소스 코드 파일을 재선택할 경우 동일한 비교 작업이 수행되고, 클라우드 모양이 변경되었다.
대응	현재 기준 소스 코드 파일로 선택되어있는 파일로 재선택할 수 없도록 수정하였다.

System Testing Respond Report

문제	ST-21. 대량의 파일을 비교시 분석 - Fail
원인	많은 양의 소스 코드 파일이 들어있는 폴더를 지정하고 시작하였을 경우 오류가 발생하였다.
대응	폴더 내에 소스 코드 파일이 100개 초과일 경우 오류 메시지를 출력하고 다시 시작 화면으로 돌아가도록 수정하였다.

문제	ST-22. 대량의 파일을 비교 후 클라우드 표현 - Fail
원인	많은 양의 소스 코드 파일이 들어있는 폴더를 지정하고 시작하였을 경우 클라우드에 비교 결과를 출력하지 못하는 오류(무한 루프)가 발생하였다.
대응	폴더 내에 소스 코드 파일이 100개 초과일 경우 오류 메시지를 출력하고 다시 시작 화면으로 돌아가도록 수정하였다. 일치율이 52% 이상인 소스 코드 파일들만 클라우드에 출력하도록 수정하였다. 클라우드 내 표현되는 텍스트의 크기를 조절하였다.

문제	ST-24. 잘못된 폴더경로 참조 - Fail
원인	실재하지 않는 잘못된 폴더 경로를 지정하였을 경우에도 분석이 시작되는 오류가 발생하였다.
대응	실재하지 않는 잘못된 폴더 경로일 경우 예외처리를 통해 오류 메시지를 출력하고 다시 시작 화면으로 돌아가도록 수정하였다.

3. Category Partition Testing

파일종류	시작하기	코드라인	주석
모든 파일이 같은 이름 다른 형식 파일	시작하기 버튼 클릭	모든 파일이 한줄코드	모든 파일이 /* */만 있는 경우
모든 파일이 같은 이름 다른 형식 파일	시작하기 버튼 클릭	모두 다 여러 라인	모든 파일이 주석이 없는 경우
모든 파일이 같은 이름 다른 형식 파일	시작하기 버튼 클릭	모든 파일이 한줄코드	모든 파일이 /* */만 있는 경우
C언어와 다른 확장자 소스파일들이 섞여있을 때	시작하기 버튼 클릭	모두 다 여러 라인	모든 파일이 /* */만 있는 경우
모든 파일이 같은 이름 다른 형식 파일	시작하기 버튼 클릭	모두 다 여러 라인	모든 파일이 /* */만 있는 경우
모든 파일이 같은 이름 다른 형식 파일	시작하기 버튼 클릭	모든 파일이 한줄코드	모든 파일이 주석이 없는 경우
모든 파일이 같은 이름 다른 형식 파일	시작하기 버튼 클릭	모두 다 여러 라인	모든 파일이 /* */만 있는 경우
모든 파일이 같은 이름 다른 형식 파일	시작하기 버튼 클릭	모든 파일이 한줄코드	모든 파일이 /**/ or // or 주석이 없는 경우
모든 파일이 같은 이름 다른 형식 파일	시작하기 버튼 클릭	모두 다 여러 라인	모든 파일이 /* */만 있는 경우
모든 파일이 같은 이름 다른 형식 파일	시작하기 버튼 클릭	모든 파일이 한줄코드	모든 파일이 //만 있는 경우
C언어와 다른 확장자 소스파일들이 섞여있을 때	시작하기 버튼 클릭	모두 다 여러 라인	모든 파일이 주석이 없는 경우
C언어와 다른 확장자 소스파일들이 섞여있을 때	시작하기 버튼 클릭	모두 다 여러 라인	모든 파일이 //만 있는 경우
모든 파일이 같은 이름 다른 형식 파일	시작하기 버튼 클릭	모두 다 여러 라인	모든 파일이 /* */만 있는 경우
모든 파일이 같은 이름 다른 형식 파일	시작하기 버튼 클릭	모두 다 여러 라인	모든 파일이 /**/ or // or 주석이 없는 경우
모든 파일이 같은 이름 다른 형식 파일	시작하기 버튼 클릭	모두 다 빈문서	모든 파일이 주석이 없는 경우
모든 파일이 같은 이름 다른 형식 파일	시작하기 버튼 클릭	파일들이 한줄코드 or 빈문서 or 여러라인 섞인 코드 라인 스타일	모든 파일이 /**/ or // or 주석이 없는 경우
C언어와 다른 확장자 소스파일들이 섞여있을 때	시작하기 버튼 클릭	모든 파일이 한줄코드	모든 파일이 /* */만 있는 경우
C언어와 다른 확장자 소스파일들이 섞여있을 때	시작하기 버튼 클릭	모든 파일이 한줄코드	모든 파일이 //만 있는 경우
C언어와 다른 확장자 소스파일들이 섞여있을 때	시작하기 버튼 클릭	모든 파일이 한줄코드	모든 파일이 //만 있는 경우
C언어와 다른 확장자 소스파일들이 섞여있을 때	시작하기 버튼 클릭	파일들이 한줄코드 or 빈문서 or 여러라인 섞인 코드 라인 스타일	모든 파일이 /**/ or // or 주석이 없는 경우
C언어와 다른 확장자 소스파일들이 섞여있을 때	시작하기 버튼 클릭	모든 파일이 한줄코드	모든 파일이 /* */만 있는 경우
C언어와 다른 확장자 소스파일들이 섞여있을 때	시작하기 버튼 클릭	모두 다 여러 라인	모든 파일이 /* */만 있는 경우
모든 파일이 같은 이름 다른 형식 파일	시작하기 버튼 클릭	모두 다 여러 라인	모든 파일이 //만 있는 경우
모든 파일이 같은 이름 다른 형식 파일	시작하기 버튼 클릭	모두 다 여러 라인	모든 파일이 /**/ or // or 주석이 없는 경우
모든 파일이 같은 이름 다른 형식 파일	시작하기 버튼 클릭	모두 다 여러 라인	모든 파일이 주석이 없는 경우
C언어와 다른 확장자 소스파일들이 섞여있을 때	시작하기 버튼 클릭	모든 파일이 한줄코드	모든 파일이 /* */만 있는 경우
C언어와 다른 확장자 소스파일들이 섞여있을 때	시작하기 버튼 클릭	모두 다 여러 라인	모든 파일이 //만 있는 경우
C언어와 다른 확장자 소스파일들이 섞여있을 때	시작하기 버튼 클릭	모든 파일이 한줄코드	모든 파일이 //만 있는 경우
C언어와 다른 확장자 소스파일들이 섞여있을 때	시작하기 버튼 클릭	모든 파일이 한줄코드	모든 파일이 주석이 없는 경우
모든 파일이 같은 이름 다른 형식 파일	시작하기 버튼 클릭	모두 다 여러 라인	모든 파일이 /**/ or // or 주석이 없는 경우
모든 파일이 같은 이름 다른 형식 파일	시작하기 버튼 클릭	모두 다 여러 라인	모든 파일이 /**/ or // or 주석이 없는 경우
모든 파일이 같은 이름 다른 형식 파일	시작하기 버튼 클릭	모든 파일이 한줄코드	모든 파일이 //만 있는 경우
모든 파일이 같은 이름 다른 형식 파일	시작하기 버튼 클릭	모든 파일이 한줄코드	모든 파일이 /* */만 있는 경우
모든 파일이 같은 이름 다른 형식 파일	시작하기 버튼 클릭	모두 다 여러 라인	모든 파일이 //만 있는 경우
C언어와 다른 확장자 소스파일들이 섞여있을 때	시작하기 버튼 클릭	모두 다 여러 라인	모든 파일이 //만 있는 경우
모든 파일이 같은 이름 다른 형식 파일	시작하기 버튼 클릭	모두 다 여러 라인	모든 파일이 /* */만 있는 경우
C언어와 다른 확장자 소스파일들이 섞여있을 때	시작하기 버튼 클릭	모든 파일이 한줄코드	모든 파일이 /**/ or // or 주석이 없는 경우
모든 파일이 같은 이름 다른 형식 파일	시작하기 버튼 클릭	모두 다 여러 라인	모든 파일이 주석이 없는 경우
모든 파일이 같은 이름 다른 형식 파일	시작하기 버튼 클릭	모든 파일이 한줄코드	모든 파일이 주석이 없는 경우
C언어와 다른 확장자 소스파일들이 섞여있을 때	시작하기 버튼 클릭	모두 다 빈문서	모든 파일이 주석이 없는 경우
모든 파일이 같은 이름 다른 형식 파일	시작하기 버튼 클릭	모두 다 여러 라인	모든 파일이 /* */만 있는 경우
모든 파일이 같은 이름 다른 형식 파일	시작하기 버튼 클릭	모두 다 여러 라인	모든 파일이 주석이 없는 경우
C언어와 다른 확장자 소스파일들이 섞여있을 때	시작하기 버튼 클릭	모든 파일이 한줄코드	모든 파일이 /* */만 있는 경우
모든 파일이 같은 이름 다른 형식 파일	시작하기 버튼 클릭	모두 다 여러 라인	모든 파일이 /* */만 있는 경우

2. 5조 System Test 대응 보고서

1. Specification

위치	Stage 1000 - Activity 1003. Define Requirements
문제	요구 사항 항목 "R.1, Select Folder, 비교할 소스 코드 파일들이 들어있는 폴더를 선택한다." 에서 "소스 코드"와 "소스 코드 파일"에 대한 정의가 불분명하다. 소스 코드는 어떤 프로그래밍 언어로 작성되어야 하는가? 해당 프로그래밍 언어의 명세 상으로 유효한, 즉 컴파일 가능한 코드여야 하는가? 소스 코드 파일의 크기에는 어떤 제한이 있는가?
대응	Stage 1000 - Activity 1004. Record Terms in Glossary에 다음과 같은 내용을 추가하였다.

Term	Description
소스 코드	컴파일 가능 여부와 상관 없이 프로그래밍 언어로 작성된 코드
소스 코드 파일	파일 크기와 상관 없이 소스 코드가 담긴 .c 확장자 파일

위치	Stage 1000 - Activity 1003. Define Requirements
문제	요구 사항 항목 "R.4, Calculate Sync–Rate, 분석하여 가공한 정보를 비교하여 일치율을 계산한다" 의 가공된 정보의 비교 방식과 일치율의 계산 기준이 명시되어있지 않다. R.4.1, R.4.2, R.4.3, R.4.4, R.4.5 도 같은 문제가 있다.

<table>
<tr><td>대응</td><td>
일치율 계산 알고리즘은 Stage 2040 - Activity 2141. Design Real Use Cases에 작성하였다.
Calculate Line Sync-Rate의 Typical Courses of Events을 아래와 같이 구체화하였다.
<table>
<tr><td>Typical Courses of Events</td><td>(A) : Actor, (S) : System
3. (S) : 기준 파일의 라인 수가 비교 파일의 라인 수보다 클 경우 '(비교 파일의 라인 수 / 기준 파일의 라인 수) * 100' 수식의 결과값을 일치율로 부여한다.
4. (S) : 일치율을 Calculate:syncRate에 저장한다.</td></tr>
<tr><td>Alternative Courses of Events</td><td>(A) : Actor, (S) : System
3. (S) : 기준 파일의 라인 수가 비교 파일의 라인 수보다 작을 경우 '(기준 파일의 라인 수 / 비교 파일의 라인 수) * 100' 수식의 결과값을 일치율로 부여한다.
4. (S) : 일치율을 Calculate:syncRate에 저장한다.</td></tr>
</table>
Calculate Function Sync-Rate의 Typical Courses of Events을 아래와 같이 구체화하였다.
<table>
<tr><td>Typical Courses of Events</td><td>(A) : Actor, (S) : System
3. (S) : 기준 파일의 함수 수가 비교 파일의 함수 수보다 클 경우 '{(비교 파일과 기준 파일에서 이름이 같은 함수 수 / 기준 파일의 함수 수) * 100} * 0.6 + {(비교 파일의 함수 수 / 기준 파일의 함수 수) * 100} * 0.4' 수식의 결과값을 일치율로 부여한다.
4. (S) : 일치율을 Calculate:syncRate에 저장한다.</td></tr>
<tr><td>Alternative Courses of Events</td><td>(A) : Actor, (S) : System
3. (S) : 기준 파일의 함수 수가 비교 파일의 함수 수보다 작을 경우 '{(비교 파일과 기준 파일에서 이름이 같은 함수 수 / 비교 파일의 함수 수) * 100} * 0.6 + {(기준 파일의 함수 수 / 비교 파일의 함수 수) * 100} * 0.4' 수식의 결과값을 일치율로 부여한다.
4. (S) : 일치율을 Calculate:syncRate에 저장한다.</td></tr>
</table>
Calculate Variable Sync-Rate의 Typical Courses of Events을 아래와 같이 구체화하였다.
<table>
<tr><td>Typical Courses of Events</td><td>(A) : Actor, (S) : System
3. (S) : 기준 파일의 변수 수가 비교 파일의 변수 수보다 클 경우 '{(비교 파일과 기준 파일에서 이름이 같은 변수 수 / 기준 파일의 변수 수) * 100} * 0.6 + {(비교 파일의 변수 수 / 기준 파일의 변수 수) * 100} * 0.4' 수식의 결과값을 일치율로 부여한다.
4. (S) : 일치율을 Calculate:syncRate에 저장한다.</td></tr>
</table>
</td></tr>
</table>

Alternative Courses of Events	(A) : Actor, (S) : System 3. (S) : 기준 파일의 변수 수가 비교 파일의 변수 수보다 작을 경우 '{(비교 파일과 기준 파일에서 이름이 같은 변수 수 / 비교 파일의 변수 수) * 100} * 0.6 + {(기준 파일의 변수 수 / 비교 파일의 변수 수) * 100} * 0.4' 수식의 결과값을 일치율로 부여한다. 4. (S) : 일치율을 Calculate:syncRate에 저장한다.

Calculate Preprocessor Sync-Rate의 Typical Courses of Events을 아래와 같이 구체화하였다.

Typical Courses of Events	(A) : Actor, (S) : System 3. (S) : 기준 파일의 전처리기 수가 비교 파일의 전처리기 수보다 클 경우 '{(비교 파일과 기준 파일에서 이름이 같은 전처리기 수 / 기준 파일의 전처리기 수) * 100} * 0.6 + {(비교 파일의 전처리기 수 / 기준 파일의 전처리기 수) * 100} * 0.4' 수식의 결과값을 일치율로 부여한다. 4. (S) : 일치율을 *Calculate:syncRate*에 저장한다.
Alternative Courses of Events	(A) : Actor, (S) : System 3. (S) : 기준 파일의 전처리기 수가 비교 파일의 전처리기 수보다 작을 경우 '{(비교 파일과 기준 파일에서 이름이 같은 전처리기 수 / 비교 파일의 전처리기 수) * 100} * 0.6 + {(기준 파일의 전처리기 수 / 비교 파일의 전처리기 수) * 100} * 0.4' 수식의 결과값을 일치율로 부여한다. 4. (S) : 일치율을 Calculate:syncRate에 저장한다.

Calculate Annotation Sync-Rate의 Typical Courses of Events을 아래와 같이 구체화하였다.

Typical Courses of Events	(A) : Actor, (S) : System 3. (S) : 기준 파일의 주석 수가 비교 파일의 주석 수보다 클 경우 '(비교 파일의 주석 수 / 기준 파일의 주석 수) * 100' 수식의 결과값을 일치율로 부여한다. 4. (S) : 일치율을 *Calculate:syncRate*에 저장한다.
Alternative Courses of Events	(A) : Actor, (S) : System 3. (S) : 기준 파일의 주석 수가 비교 파일의 주석 수보다 작을 경우 '(기준 파일의 주석 수 / 비교 파일의 주석 수) * 100' 수식의 결과값을 일치율로 부여한다. 4. (S) : 일치율을 Calculate:syncRate에 저장한다.

System Testing Respond Report

위치	Stage 1000 - Activity 1001. Define Draft Plan
문제	"A. 설명이 필요 없을 정도의 조작하기 쉬운 인터페이스" 에서 주관적인 어휘가 다수 사용되었으며, 기준이 불분명하다. "B. 한눈에 이해 가능한 결과 출력" 상동. "C. 다른 언어의 소스 파일도 비교 가능하도록 하는 확장성" 에서 "다른" 의 기준이 되는 언어가 명세되어 있지 않다.
대응	Non-Function Requirements의 내용을 아래와 같이 구체화하였다. A. 조작하기 쉬운 인터페이스 B. 한 눈에 이해 가능한 결과 출력 C. C 소스 코드 파일만 비교 D. 마우스 왼쪽 클릭으로만 조작

위치	Stage 1000 - Activity 1010. Refine Plan
문제	"A. 소스 코드 파일 분석 작업은 5초 이내로 수행되어야 한다" 고 명세되어 있는데, 분석 대상이 되는 코드의 최대량 또는 기준량이 명시되어 있지 않았으며, "B. 기준 소스 코드 파일 변경 시 2초 이내로 비교 결과가 변경되어야 한다." 역시 같은 문제가 있다.
대응	Activity 1007. Define Business Use Case에 정의한 '메인 화면에서의 Actor의 행동'에 '비교하고자 하는 100개 이하인 소스 코드 파일들이 들어있는 폴더를 선택하고, '시작하기' 버튼을 클릭하여 유사한 소스 코드 파일이 있는지 비교할 수 있다.'와 같이 소스 코드 파일의 최대량을 100개로 설정하였다.

위치	Stage 1000 - Activity 1010. Refine Plan
문제	하나의 바이너리로 여러 운영 체제에서 실행될 수 있는 Java 프로그래밍 언어를 채택하였는데 "Microsoft Windows 7 이상" 의 실행 환경만을 지원한다.
대응	안정적인 프로그램 이용을 위해서는 개발 환경과 테스트 환경이 비슷한 Microsoft Windows 7 이상의 운영체제를 권장한다.

위치	Stage 1000 - Activity 1003. Define Requirements
문제	"B.4, 종료 버튼, 화면 하단에 위치하여 클릭 시 프로그램이 종료된다", "C. 그 외의 메시지 출력은 대화상자(Dialog) 활용" 이라고 명세되어 있는데, 종료 버튼과 대화 상자가 없다.
대응	종료 버튼에 관한 내용을 삭제하였고, 메시지 출력은 대화상자(Dialog)를 활용하도록 수정하였다. Other Requirements에 '새로운 폴더를 지정하기 위해서는 프로그램을 종료하고 다시 실행해야 한다.'는 내용을 추가하였다.

위치	Stage 1000 - Activity 1003. Define Requirements
문제	"A. 비교 결과(일치율, 유사항목)를 한 눈에 알아볼 수 있도록 비교 결과 출력 화면이 직관적이어야 한다." 에서 직관성의 판단 기준이 명시되어 있지 않다.
대응	해당 내용이 모호하다고 판단하여 삭제하였다.

2. Brute Force Testing

문제	메인 인터페이스 창을 최소화 하였다 복구시키니 사용자 인터페이스가 육안으로 확인할 수 없을 정도로 작아지며 원상 복구되지 않는다.
원인	동일한 작업을 100번 수행한 결과 이상이 없었으므로 테스트 환경으로 인한 문제로 생각된다.
대응	N/A

문제	명세서 상 비교 결과 일치율 85% 이상인 파일은 클론으로 판정하고, 이 결과를 한 눈에 알아볼 수 있도록 직관적이어야 하는데 클론 판정 유무가 인터페이스에 드러나지 않는다.
원인	클라우드 내 출력되는 텍스트의 크기가 작아 직관적으로 결과를 인식할 수 없었다.
대응	보고서 상 Clone의 정의를 삭제하였다. 클라우드 내 출력되는 텍스트의 크기를 조절하였다. Clone에 대한 판정이 아닌 사용자의 Clone 판정에 도움을 주는 프로그램의 용도로 수정하였다.

문제	/* comment */ 스타일의 주석을 사용한 코드를 분석하였을 때 클론 체커가 멈춘다.
원인	동일한 작업을 수행한 결과 이상이 없었으므로 테스트 환경으로 인한 문제로 생각된다.
대응	N/A

문제	세미콜론을 사용해 코드 여러 줄을 합쳤을 때 "Use Case: Divide Code Line"기능이 작동하지 않는다.
원인	한 줄에 같이 작성된 코드들의 행을 나누는 divideCodeLine() 메소드 내에서 무한 루프가 발생되었다.
대응	divideCodeLine() 메소드의 알고리즘을 수정하였다.