

Software Verification

2st Testing - System Test & Static Analysis

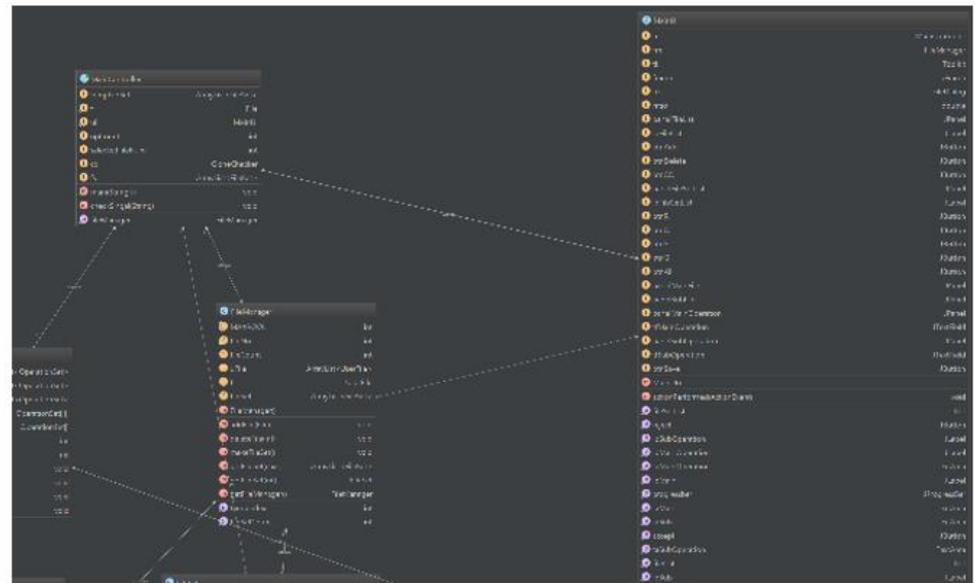
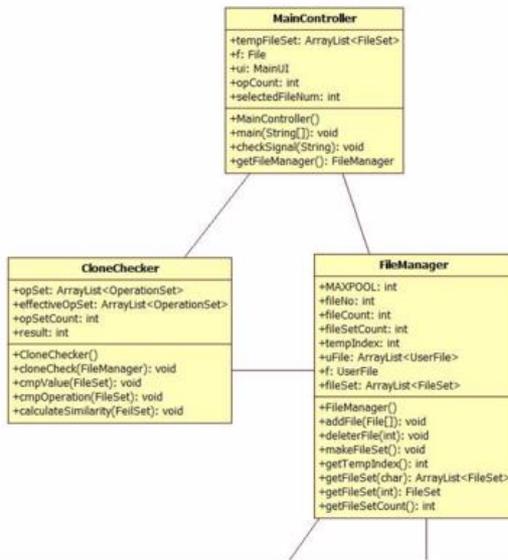
2조

이상혁

왕홍강

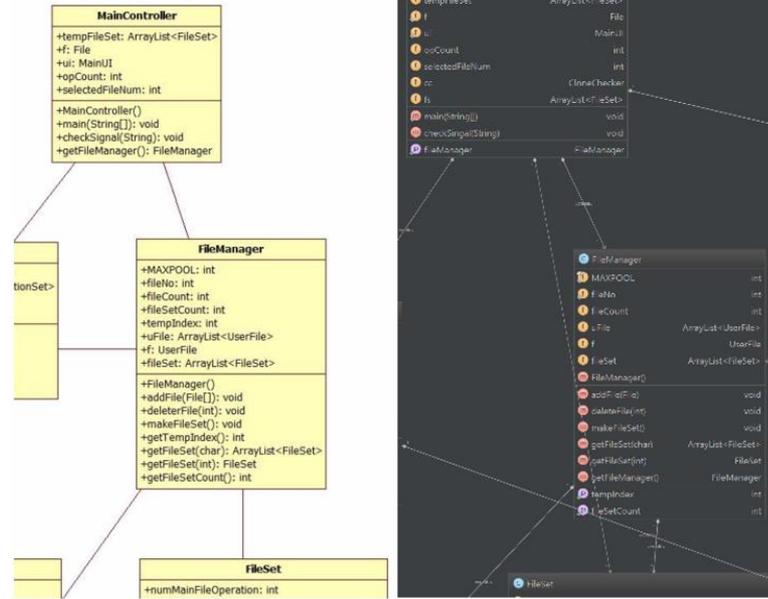
김태영

Specification and code review



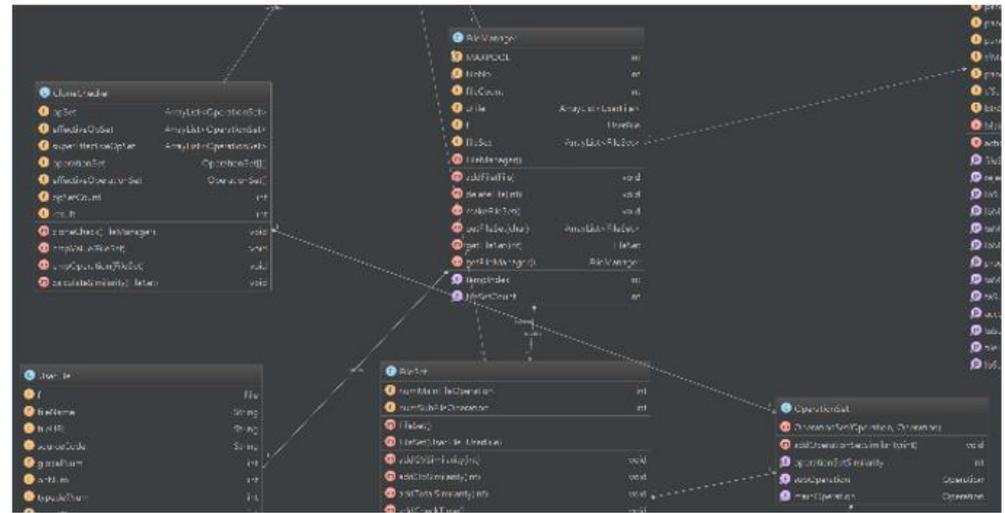
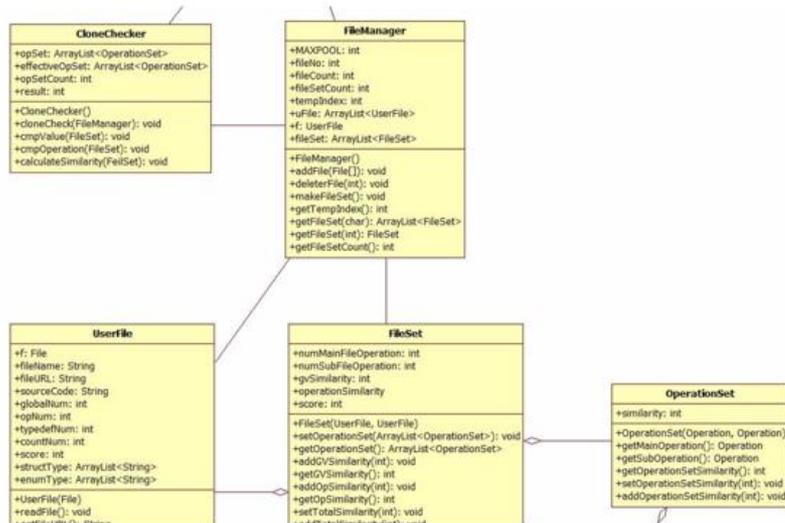
문서에 존재하지 않던 MainUI가 등장

Specification and code review



문서에 존재하지 않던 MainController과 FileSet 사이의 관계가 있음

Specification and code review



문서와 다르게 구현상에는 CloneChecker와 OperationSet의 관계가 있으며, CloneChecker와 FileManager와의 관계가 없다.

Specification and code review

메소드와 변수의 경우에도 문서에 누락된 내용이 상당 수 존재

```
MainController
├── tempFileSet      ArrayList<FileSet>
├── f                File
├── ui              MainUI
├── opCount         int
├── selectedFileNum int
├── cc              CloneChecker
├── fs              ArrayList<FileSet>
├── main(String[]) void
├── checkSingal(String) void
└── fileManager    FileManager

getFileManager 누락
```

```
CloneChecker
├── opSet           ArrayList<OperationSet>
├── effectiveOpSet ArrayList<OperationSet>
├── superEffectiveOpSet ArrayList<OperationSet>
├── operationSet   OperationSet[][]
├── effectiveOperationSet OperationSet[]
├── opSetCount     문서에 누락 int
├── result         int
├── cloneCheck(FileManager) void
├── cmpValue(FileSet) void
├── cmpOperation(FileSet) void
└── calculateSimilarity(FileSet) void
```

Specification and code review

메소드와 변수의 경우에도 문서에 누락된 내용이 상당 수 존재

Symbol	Member Name	Type	Annotation
C	FileManager		
I	MAXPOOL	int	
I	fileNo	int	
I	fileCount	int	
I	uFile	ArrayList<UserFile>	
I	f	UserFile	
I	fileSet	ArrayList<FileSet>	
M	FileManager()		
M	addFile(File)	void	문서에는 File[]
M	deleteFile(int)	void	
M	makeFileSet()	void	
M	getFileSet(char)	ArrayList<FileSet>	getTempIndex 누락
M	getFileSet(int)	FileSet	
M	getManager()	FileManager	getFileSetCount 누락
P	tempIndex	int	
P	fileSetCount	int	

Symbol	Member Name	Type	Annotation
C	UserFile		
I	f	File	
I	fileName	String	
I	fileURL	String	
I	sourceCode	String	
I	globalNum	int	
I	opNum	int	
I	typedefNum	int	
I	countNum	int	
I	structType	ArrayList<String>	
I	enumType	ArrayList<String>	
M	UserFile(File)		
M	UserFile()		
M	readFile()	void	getFileSetName 누락
M	checkSyntax()	void	getSourceCode 누락
M	checkGlobalValue(String)	void	getScore 누락
M	checkOperation(Operation)	void	getGlobalValue 누락
M	checkBranch(Operation)	void	getOperation 누락
M	checkParameter(Operation)	void	
M	checkFunction(Operation)	void	
M	calculateScore()	void	
P	globalValue	ArrayList<GlobalValue>	
P	fileName	String	
P	sourceCode	String	
P	score	int	
P	fileURL	String	
P	operation	ArrayList<Operation>	

Specification and code review

메소드와 변수의 경우에도 문서에 누락된 내용이 상당 수 존재

FileSet

- numMainFileOperation int
- numSubFileOperation int
- FileSet()
- FileSet(UserFile, UserFile)
- addGVSimilarity(int) void
- addOpSimilarity(int) void
- addTotalSimilarity(int) void
- addCheckTime() void
- subFile UserFile
- GVSimilarity int
- mainFile UserFile
- operationSet ArrayList<OperationSet>
- checkTime int
- opSimilarity int
- score int
- maxScore int
- fileSetName String
- totalSimilarity int

Annotations:

- setOperationSet 누락
- getOperationSet 누락
- 들 문서에 있는
- getter/setter 전부 누락
- gvSimilarity 로 되어있음 (소문자)
- operationSimilarity로 정의됨
- 문서에 누락 됨

OperationSet

- OperationSet(Operation, Operation)
- addOperationSetSimilarity(int) void
- operationSetSimilarity int
- subOperation Operation
- mainOperation Operation

OperationSet

- +similarity: int
- +OperationSet(Operation, Operation)
- +getMainOperation(): Operation
- +getSubOperation(): Operation
- +getOperationSetSimilarity(): int
- +setOperationSetSimilarity(int): void
- +addOperationSetSimilarity(int): void

Annotation: 심하게 다름

Specification and code review

메소드와 변수의 경우에도 문서에 누락된 내용이 상당 수 존재

GlobalValue

- +name: String
- +globalValueType: int
- +useCount: int
- +stdio: boolean
- +stdlib: boolean
- +time: boolean
- +windows: boolean
- +string: boolean
- +conio: boolean

- +GlobalValue()
- +setHeader(String): void
- +getStdio(): boolean
- +getStdlib(): boolean
- +getTime(): boolean
- +getWindows(): boolean
- +getString(): boolean
- +getConio(): boolean
- +setGlobalValueType(int): void
- +getGlobalValueType(): int
- +setGlobalValueName(String): void
- +getGlobalValueName(): String
- +setUseCount(int): void
- +getUseCount(): int
- +getGlobalValue(): GlobalValue

getter/setter 누락
name, string 등 변
수 누락

Operation

- numLocalValue: int
- localIndex: int
- score: int

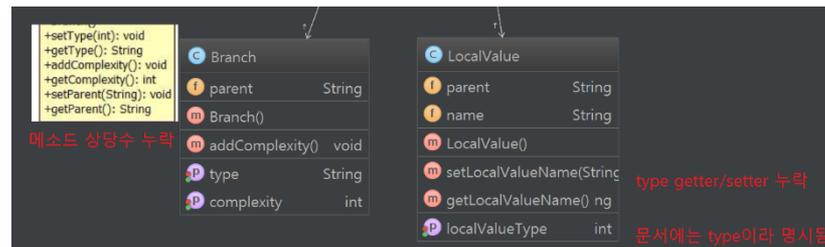
- +Operation()
- +addNumFunction(): void
- +addLocalValue(int, String): void
- +addBranch(): void
- +addLineNum(): void
- +name: String
- +source: String
- +branch: ArrayList<Branch>
- +numFunction: int
- +lineNum: int
- +function: int[]
- +localValue: ArrayList<LocalValue>
- +returnType: int
- +parameterArray: int[]
- +parameter: String
- +score: int

- +Operation()
- +setReturnType(int): void
- +getReturnType(): int
- +setName(String): void
- +getName(): String
- +setParameter(String): void
- +getParameter(): void
- +getParameterArray(): int[]
- +setSource(): String
- +addNumFunction(): void
- +getNumFunction(): int
- +getFunction(): int[]
- +addLocalValue(int, String): void
- +getLocalValue(): ArrayList<LocalValue>
- +addBranch(): void
- +getBranch(): ArrayList<Branch>

차이가 심함...

Specification and code review

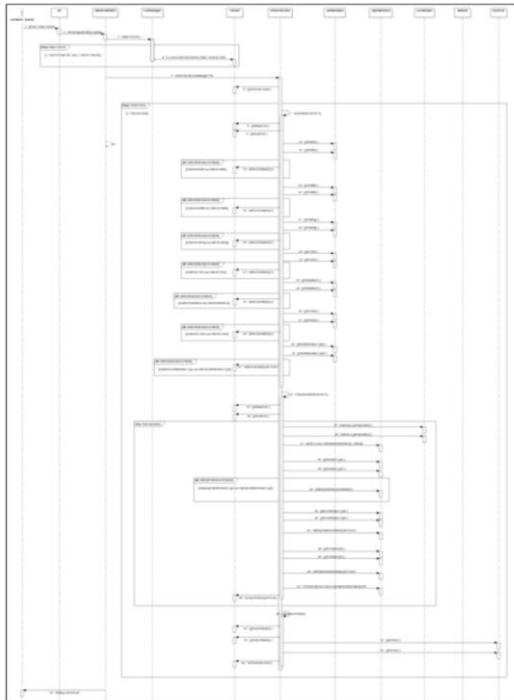
메소드와 변수의 경우에도 문서에 누락된 내용이 상당 수 존재



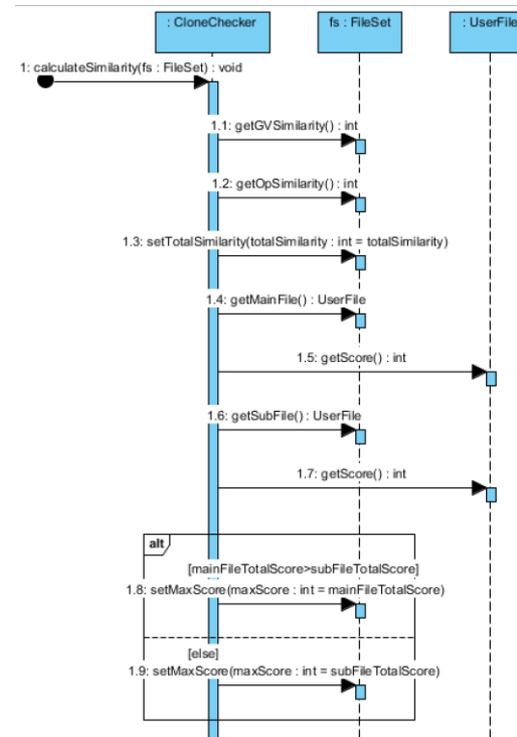
Specification and code review

sequence diagram

(메소드 누락 등으로 작업에 어려움이 큼)



[그림4] Clone Check



Brute Force Testing

7	Clone Check 버튼 클릭	T
8	Clone Check 버튼을 여러 번 클릭	F
9	Show Red 버튼 클릭	T



7	Clone Check 버튼 클릭	T
8	Clone Check 버튼을 여러 번 클릭	T
9	Show Red 버튼 클릭	T

16	Save 버튼 클릭	T
17	동일 파일을 두 번 오픈	F

Category Partitioning Testing

Test Case No.	Pass
101,111,121,202	T
101,112,121,202	T
102,111,121,201,301,401,411,421	F
102,111,121,201,301,401,412,421	F
102,111,121,201,302,401,411,421	F
102,111,121,201,302,401,412,421	F
102,111,121,201,303,401,411,421	F
102,111,121,201,303,401,412,421	F
102,111,121,201,304,401,411,421	F
102,111,121,201,304,401,412,421	F
102,111,121,201,305,401,411,421	F
102,111,121,201,305,401,412,421	F
102,112,121,202	T



Test Case No.	Pass
101,112,121,202	T
102,111,121,201,301,401,411,421	T
102,111,121,201,301,401,412,421	T
102,111,121,201,302,401,411,421	T
102,111,121,201,302,401,412,421	T
102,111,121,201,303,401,411,421	T
102,111,121,201,303,401,412,421	T
102,111,121,201,304,401,411,421	T
102,111,121,201,304,401,412,421	T
102,111,121,201,305,401,411,421	T
102,111,121,201,305,401,412,421	T

Pairwise Combination Testing

Pair		Pass
{101, 202}	파일 하나를 추가한 뒤에 클론 체크 버튼을 클릭하면 클론 체크가 되지 않는다.	P
{102, 201}	동일 파일을 두개 넣은뒤 <u>클론체크</u> 를 누르면 100%가 출력되어야 한다.	F
{103, 112, 202}	서로 다른 소스파일을 두개 추가하고 하나를 삭제 한뒤에 클론 체크 버튼을 클릭하면 동작하지 않아야 한다.	P
{107, 111, 301}	서로 다른 소스파일을 10개 추가한 후 아무것도 클릭하지 않고 삭제버튼을 클릭한다 이후 Show Red 버튼을 클릭한다.	P
{107, 111, 302}	서로 다른 소스파일을 10개 추가한 후 아무것도 클릭하지 않고 삭제버튼을 클릭한다 이후 Show Orange 버튼을 클릭한다.	P
{107, 111, 303}	서로 다른 소스파일을 10개 추가한 후 아무것도 클릭하지 않고 삭제버튼을 클릭한다 이후 Show Yellow 버튼을 클릭한다.	P
{107, 111, 304}	서로 다른 소스파일을 10개 추가한 후 아무것도 클릭하지 않고 삭제버튼을 클릭한다 이후 Show Green 버튼을 클릭한다.	P
{107, 111, 305}	서로 다른 소스파일을 10개 추가한 후 아무것도 클릭하지 않고 삭제버튼을 클릭한다 이후 Show ALL 버튼을 클릭한다.	P
{401, 411, 421}	<u>유사도</u> 를 확인하고 Accept 버튼 클릭을 클릭한 뒤 Save 버튼을 클릭하면 File Set에 변경된 유사도가 적용됨.	P
{401, 412, 421}	<u>유사도</u> 를 확인하고 Reject 버튼 클릭을 클릭한 뒤 Save 버튼을 클릭하면 File Set에 변경된 유사도가 적용됨.	P



Pair		Pass
{101, 202}	파일 하나를 추가한 뒤에 클론 체크 버튼을 클릭하면 클론 체크가 되지 않는다	T
{102, 201}	동일 파일을 두개 넣은뒤 클론체크를 누르면 100%가 출력되어야 한다	T
{103, 112, 202}	서로 다른 소스파일을 두개 추가하고 하나를 삭제 한뒤에 클론 체크 버튼을 클릭하면 동작하지 않아야 한다	T
{107, 111, 301}	서로 다른 소스파일을 10개 추가한 후 아무것도 클릭하지 않고 삭제버튼을 클릭한다 이후 Show Red 버튼을 클릭한다	T
{107, 111, 302}	서로 다른 소스파일을 10개 추가한 후 아무것도 클릭하지 않고 삭제버튼을 클릭한다 이후 Show Orange 버튼을 클릭한다	T
{107, 111, 303}	서로 다른 소스파일을 10개 추가한 후 아무것도 클릭하지 않고 삭제버튼을 클릭한다 이후 Show Yellow 버튼을 클릭한다	T
{107, 111, 304}	서로 다른 소스파일을 10개 추가한 후 아무것도 클릭하지 않고 삭제버튼을 클릭한다 이후 Show Green 버튼을 클릭한다	T
{107, 111, 305}	서로 다른 소스파일을 10개 추가한 후 아무것도 클릭하지 않고 삭제버튼을 클릭한다 이후 Show ALL 버튼을 클릭한다	T
{401, 411, 421}	유사도를 확인하고 Accept 버튼 클릭을 클릭한 뒤 Save 버튼을 클릭하면 File Set에 변경된 유사도가 적용됨	T
{401, 412, 421}	유사도를 확인하고 Reject 버튼 클릭을 클릭한 뒤 Save 버튼을 클릭하면 File Set에 변경된 유사도가 적용됨	T

Static Analysis

Checkstyle

구글 스타일 기준 전체 3759개의 에러

Coding Style Check Results

Summary	
Total files checked	11
Files with errors	11
Total errors	3759
Errors per file	342

File:	C:\Users\shlee322\projects\konkuk-dslab\src\main\java\WController\WCloneChecker.java
Line Number	Error Message
1	Package name 'Controller' must match pattern '^[a-z]+(\W[a-z][a-z0-9]*)*\$'.
2	'import' should be separated from previous statement.
4	Import statement for 'Model.*' is in the wrong order. Should be in the 'THIRD_PARTY_PACKAGE' group, expecting not assigned imports on this line.
4	Using the '*.*' form of import should be avoided - Model.*.
8	'member def type' have incorrect indentation level 3, expected level should be 2.
9	'member def type' have incorrect indentation level 3, expected level should be 2.
10	'member def type' have incorrect indentation level 3, expected level should be 2.
11	'member def type' have incorrect indentation level 3, expected level should be 2.
11	Array brackets at illegal position.
12	'member def type' have incorrect indentation level 3, expected level should be 2.
12	Array brackets at illegal position.
13	'member def type' have incorrect indentation level 3, expected level should be 2.
14	'member def type' have incorrect indentation level 3, expected level should be 2.
16	'method def modifier' have incorrect indentation level 3, expected level should be 2.
17	'method def' child have incorrect indentation level 6, expected level should be 4.
19	'for' have incorrect indentation level 6, expected level should be 4.
20	'for' child have incorrect indentation level 9, expected level should be 6.
21	'for' child have incorrect indentation level 9, expected level should be 6.
22	'for' child have incorrect indentation level 9, expected level should be 6.
23	'for rcurly' have incorrect indentation level 6, expected level should be 4.
24	'method def rcurly' have incorrect indentation level 3, expected level should be 2.
26	'method def modifier' have incorrect indentation level 3, expected level should be 2.
27	'method def' child have incorrect indentation level 6, expected level should be 4.
27	WhitespaceAround: '+' is not preceded with whitespace.

Static Analysis

Findbug

Metrics

1819 lines of code analyzed, in 11 classes, in 3 packages.

Metric	Total	Density*
High Priority Warnings	13	7.15
Medium Priority Warnings	31	17.04
Total Warnings	44	24.19

Warning Type	Number
Bad practice Warnings	12
Correctness Warnings	3
Experimental Warnings	1
Malicious code vulnerability Warnings	2
Performance Warnings	18
Dodgy code Warnings	8
Total	44

Static Analysis

Findbug

Bad practice Warnings

Code	Warning
DMI	removeAll used to clear a collection in Controller.CloneChecker.cmpOperation(FileSet)
ES	Comparison of String parameter using == or != in Controller.MainController.checkSingal(String)
ES	Comparison of String parameter using == or != in Controller.MainController.checkSingal(String)
ES	Comparison of String parameter using == or != in Controller.MainController.checkSingal(String)
ES	Comparison of String parameter using == or != in Controller.MainController.checkSingal(String)
ES	Comparison of String parameter using == or != in Controller.MainController.checkSingal(String)
ES	Comparison of String parameter using == or != in Controller.MainController.checkSingal(String)
ES	Comparison of String parameter using == or != in Controller.MainController.checkSingal(String)
ES	Comparison of String parameter using == or != in Controller.MainController.checkSingal(String)
ES	Comparison of String parameter using == or != in Controller.MainController.checkSingal(String)
ES	Comparison of String parameter using == or != in Controller.MainController.checkSingal(String)
ES	Comparison of String parameter using == or != in Controller.MainController.checkSingal(String)

```
public void checkSingal(String signal) {
    if(signal == "del") {
        fm.deleteFile(ui.getFileList().getSelectedIndex());
        ui.getFileList().remove(ui.getFileList().getSelectedIndex());
    }
    else if(signal == "cc") {
        fm.makeFileSet();
        ui.getFileSetList().removeAll();
        ui.getFileSetList().clear();
        fs = fm.getFileSet('a');

        cc.cloneCheck(this.fm);
        for(int i=0; i<fm.getFileSetCount(); i++) {
            ui.getFileSetList().add(fs.get(i).getFileSetName());
        }
    }
    else if(signal == "red") {
        ui.getFileSetList().removeAll();
        ui.getFileSetList().clear();
        fs = fm.getFileSet('r');
        for(int i=0; i<fm.getTempIndex(); i++) {
            ui.getFileSetList().add(fs.get(i).getFileSetName());
        }
    }
}
```

Static Analysis

JDepend

Package	Total Classes	<u>Abstract Classes</u>	<u>Concrete Classes</u>	<u>Afferent Couplings</u>	<u>Efferent Couplings</u>	<u>Abstractness</u>	<u>Instability</u>	<u>Distance</u>
<u>Controller</u>	3	0	3	1	7	0	0.88	0.12
<u>Model</u>	7	0	7	1	3	0	0.75	0.25
<u>View</u>	1	0	1	1	6	0	0.86	0.14

Static Analysis

PMD

Avoid using if statements without curly braces : 6개

These nested if statements could be combined : 4개

Avoid empty if statements : 1개

Use equals() to compare strings instead of '==' or '!=' : 11개

The String literal "END" appears 6 times in this file; : 1개

The String literal "Line:" appears 4 times in this file; : 1개

Useless parentheses. : 2개

The String literal "consola" appears 6 times in this file; : 1개

등 약 40개 존재

[The String literal "float" appears 6 times in this file; the first occurrence is on line 202](#)

[The String literal "int" appears 6 times in this file; the first occurrence is on line 202](#)

[The String literal "long" appears 6 times in this file; the first occurrence is on line 202](#)

[The String literal "short" appears 6 times in this file; the first occurrence is on line 202](#)

[Avoid instantiating String objects; this is usually unnecessary.](#)

[Avoid instantiating String objects; this is usually unnecessary.](#)

[These nested if statements could be combined](#)

[Useless parentheses.](#)

[Useless parentheses.](#)

[The String literal "consola" appears 6 times in this file; the first occurrence is on line 169](#)

Static Analysis

jUnit & Test Code Coverage

Test Summary

58	0	0	1.597s
tests	failures	ignored	duration

100%
successful

Packages

Classes

Class	Tests	Failures	Ignored	Duration	Success rate
Controller.FileManagerTest	9	0	0	0.007s	100%
Model.BranchTest	5	0	0	0s	100%
Model.FileSetTest	15	0	0	0.012s	100%
Model.GlobalValueTest	15	0	0	0.002s	100%
Model.UserFileTest	14	0	0	1.576s	100%

Static Analysis

jUnit & Test Code Coverage

COVERAGE	FILE	LINES	RELEVANT	COVERED	MISSED	HITS/LINE
0.0	...k-dslab/src/main/java/Controller/CloneChecker.java	403	238	0	238	0.0
0.0	...konkuk-dslab/src/main/java/Model/OperationSet.java	34	12	0	12	0.0
0.0	...dslab/src/main/java/Controller/MainController.java	166	105	0	105	0.0
0.0	...lee322/konkuk-dslab/src/main/java/View/MainUI.java	352	225	0	225	0.0
40.9	...322/konkuk-dslab/src/main/java/Model/UserFile.java	1288 +36	775 +23	317 +317	458 -294	32232.0 +32232
49.12	...uk-dslab/src/main/java/Controller/FileManager.java	118	57	28 +28	29 -28	716.0 +716
65.12	...e322/konkuk-dslab/src/main/java/Model/FileSet.java	110	43	28 +11	15 -11	2125.0 +2122
81.82	...2/konkuk-dslab/src/main/java/Model/LocalValue.java	30	11	9 +9	2 -9	916.0 +916
84.62	...ee322/konkuk-dslab/src/main/java/Model/Branch.java	33	13	11	2	1367.0 +1363
88.0	...22/konkuk-dslab/src/main/java/Model/Operation.java	119	50	44 +44	6 -44	1128.0 +1128
100.0	.../konkuk-dslab/src/main/java/Model/GlobalValue.java	108	41	41 +6	0 -6	27.0 +22

전체 코드의 30.45%만 테스트 됨

Static Analysis

jUnit & Test Code Coverage

```
69         }
70         if(c != '/' && c != '*') {
71             sourceCode += '/';
72             while(true) {
73                 sourceCode += (char) data;
74                 if((data = input.read()) == 10) {
75                     break;
76                 }
77             }
78             break;
79         }
80     }
81     sourceCode += '\n';
82 }
83 else {
84     sourceCode += (char) data;
85 }
86 }
87 input.close();
88 } catch (FileNotFoundException e) {
89     e.printStackTrace();
90 } catch (IOException e) {
91     e.printStackTrace();
92 }
93 this.checkSyntax();
```

```
71         if(fileSet.get(i).getScore()>=70){
72             tempFileSetList.add(fileSet.get(i));
73             this.tempIndex++;
74         }
75     }
76     return tempFileSetList;
77 }else if(degree=='o') {
78     for(int i=0; i<fileSetCount; i++) {
79         if(fileSet.get(i).getScore()>=50 && fileSet.get(i).getScore()<70) {
80             tempFileSetList.add(fileSet.get(i));
81             this.tempIndex++;
82         }
83     }
84     return tempFileSetList;
85 }else if(degree=='y') {
86     for(int i=0; i<fileSetCount; i++) {
87         if(fileSet.get(i).getScore()>=30 && fileSet.get(i).getScore()<50) {
88             tempFileSetList.add(fileSet.get(i));
89             this.tempIndex++;
90         }
91     }
92     return tempFileSetList;
93 }else if(degree=='g') {
94     for(int i=0; i<fileSetCount; i++) {
95         if(fileSet.get(i).getScore()<30){
96             tempFileSetList.add(fileSet.get(i));
97             this.tempIndex++;
98         }
99     }
100     return tempFileSetList;
101 }else if(degree=='a'){
102     return fileSet;
103 }
```