

# Software Verification

## Static Analysis Report for Team 8

Date 2019-06-07

Team 4

컴퓨터공학과 201411257 강정모

컴퓨터공학과 201511269 송상연

컴퓨터공학과 201511270 신승화

1. SonarQube
  - 1.1 URL
  - 1.2 Overview

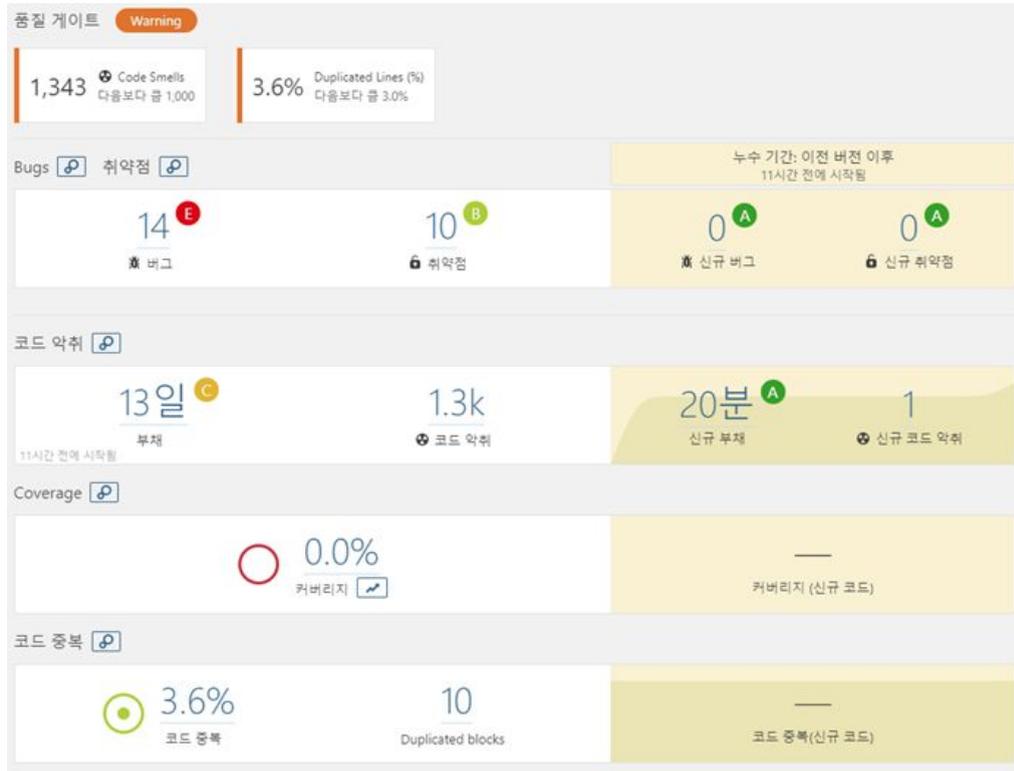
- 1.3 Bugs
- 1.4 Vulnerability
- 1.5 Code Smells
- 1.6 Coverage
  - 1.6.1 IntelliJ - Junit
  - 1.5.2 Eclipse - EclEmma
- 1.7 Duplications
- 1.8 Summary

## 2. SonarQube

### 1.1 URL

<http://ec2-15-164-99-14.ap-northeast-2.compute.amazonaws.com:9000/dashboard?id=SMA08>

## 1.2 Overview



Type	Count
Bug	14
Vulnerability	10
Code Smell	1.3k

- Bug			
심각도	Count	심각도	Count
Blocker	1	Minor	2
Critical	0	Info	0
Major	11		

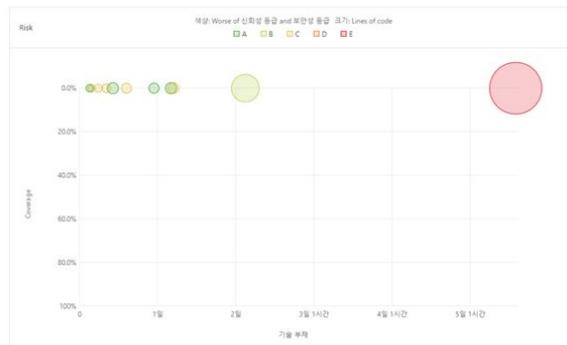
  

- Vulnerability			
심각도	Count	심각도	Count
Blocker	0	Minor	10
Critical	0	Info	0
Major	0		

- Code smell			
심각도	Count	심각도	Count
Blocker	1	Minor	1.1k
Critical	52	Info	0
Major	174		

### Project overview



## 1.3 Bugs(14개)

- Blocker(1개)

[digital-watch/src/main/java/SystemUI.java](#)

□ Add an end condition to this loop. ... 11일 전 L1443 cert

🔗 Bug 🚫 Blocker 🔵 Open 📄 할당되지 않음 ⏱ 15min effort 💬 코멘트

- Major(11개)

digital-watch/src/main/java/Alarm.java

**Either re-interrupt this method or rethrow the "InterruptedException".** ... 11일 전 L127   
 Bug Major Open 할당되지 않음 15min effort 코멘트 cwe, error-handling, multi-threading

digital-watch/src/main/java/CreateMode.java

**Class CreateMode defines fields that are used only as locals** ... 18일 전 L5   
 Bug Major Open 할당되지 않음 코멘트 correctness

**Unread field: CreateMode.createList** ... 18일 전 L5   
 Bug Major Open 할당되지 않음 30min effort 코멘트 performance

digital-watch/src/main/java/DeleteMode.java

**Class DeleteMode defines fields that are used only as locals** ... 19일 전 L5   
 Bug Major Open 할당되지 않음 코멘트 correctness

**Unread field: DeleteMode.deleteList** ... 19일 전 L5   
 Bug Major Open 할당되지 않음 30min effort 코멘트 performance

digital-watch/src/main/java/ModeSelector.java

**Class ModeSelector defines fields that are used only as locals** ... 18일 전 L1   
 Bug Major Open 할당되지 않음 코멘트 correctness

**Unused field: ModeSelector.createList** ... 18일 전 L1   
 Bug Major Open 할당되지 않음 30min effort 코멘트 performance

**Unused field: ModeSelector.deleteList** ... 18일 전 L1   
 Bug Major Open 할당되지 않음 30min effort 코멘트 performance

digital-watch/src/main/java/SystemUI.java

**Either re-interrupt this method or rethrow the "InterruptedException".** ... 13일 전 L1440   
 Bug Major Open 할당되지 않음 15min effort 코멘트 cwe, error-handling, multi-threading

digital-watch/src/main/java/Timer.java

**Either re-interrupt this method or rethrow the "InterruptedException".** ... 12일 전 L27   
 Bug Major Open 할당되지 않음 15min effort 코멘트 cwe, error-handling, multi-threading

- Minor(2개)

digital-watch/src/main/java/SystemUI.java

**Use "Thread.currentThread().getContextClassLoader()" instead.** ... 6일 전 L1072   
 Bug Minor Open 할당되지 않음 5min effort 코멘트 jee

**Use "Thread.currentThread().getContextClassLoader()" instead.** ... 6일 전 L1366   
 Bug Minor Open 할당되지 않음 5min effort 코멘트 jee

분석 예시.

```

    }
    Thread.sleep(500);
} catch (InterruptedException e) {break;}

```

Either re-interrupt this method or rethrow the "InterruptedException". ... 11일 전 ▾ L127 🔗

🐛 Bug ▾ 🚨 Major ▾ 🔓 Open ▾ 할당되지 않음 ▾ 15min effort 코멘트

🔗 cwe, error-handling, multi-threading ▾

```

    Thread.sleep(1000);
} catch (InterruptedException e) {}

```

Either re-interrupt this method or rethrow the "InterruptedException". ... 12일 전 ▾ L27 🔗

🐛 Bug ▾ 🚨 Major ▾ 🔓 Open ▾ 할당되지 않음 ▾ 15min effort 코멘트

🔗 cwe, error-handling, multi-threading ▾

```

    }
    try {
        t.sleep(10);
    } catch (InterruptedException e) {

```

Either re-interrupt this method or rethrow the "InterruptedException". ... 14일 전 ▾ L1440 🔗

🐛 Bug ▾ 🚨 Major ▾ 🔓 Open ▾ 할당되지 않음 ▾ 15min effort 코멘트

🔗 cwe, error-handling, multi-threading ▾

```

        e.printStackTrace();
    }
}

```

→ interruptException 문제가 발생하였다.  
 - 현재 사용하는 thread에게 확실하게 interrupt를 발생시켜 주어야 한다.

#### 1.4 Vulnerability(10개)

- 접근 지정자 문제가 존재함.
- 값이 변경될 수 있는 취약점이 존재함.

분석 예시.

```

public void setSettingModeList(ArrayList<String> settingModeList){
    this.settingModeList=settingModeList;
}

```

Store a copy of "settingModeList". ... 19일 전 ▾ L45 🔗

🔒 Vulnerability ▾ 🟢 Minor ▾ 🔓 Open ▾ 할당되지 않음 ▾ 5min effort 코멘트

🔗 cert, cwe, unpredictable ▾

```

// createMode.setCreatelList(createList);
// deleteMode.setDeletelList(deleteList);
}

```

→ copy store 문제가 발생한다.  
 - \*.clone() method를 사용해야 한다.

#### 1.5 Code Smells(1.3k)

SonarQube URL 참고.

분석 예시.

```
public class DeleteMode {  
    // private String[] deletelist;  
    1 private ArrayList<String> deletelist;  
}
```

Add a constructor to the class, or provide default values. ... 18일 전 L6  
Code Smell Major Open 활동되지 않음 5min effort 코멘트 pitfall

```
public class CreateMode {  
    // private String[] createlist;  
    1 private ArrayList<String> createlist;  
  
    // 현재 생성해야하는 모드들이 createlist에 저장  
    public void setCreatelist(ArrayList<String> createlist) {  
        this.createlist = createlist;  
        createModeByList();  
    }  
}
```

Add a constructor to the class, or provide default values. ... 18일 전 L4  
Code Smell Major Open 활동되지 않음 5min effort 코멘트 pitfall

→ 생성자를 사용하거나, default value를 할당해주어야 한다.

```
public void setStopwatch(String time) {  
    String[] times = time.split("\\s");  
}
```

Rename "times" which hides the field declared at line 15. ... 17일 전 L74  
Code Smell Major Open 활동되지 않음 5min effort 코멘트 based-on-misra, cert, pitfall, suspicious

```
for (int i = 0; i < matchTable.length; i++) {  
    dt[i] = 0;  
    for (int j = 0; j < 12; j++) {  
    }
```

Rename "i" which hides the field declared at line 8. ... 13일 전 L343  
Code Smell Major Open 활동되지 않음 5min effort 코멘트 based-on-misra, cert, pitfall, suspicious

Rename "j" which hides the field declared at line 8. ... 13일 전 L345  
Code Smell Major Open 활동되지 않음 5min effort 코멘트 based-on-misra, cert, pitfall, suspicious

→ 같은 이름을 갖는 다른 변수가 code 상에 존재한다.

```
lblFirst.setText("Timekeeping");  
t.start();
```

Move this "start" call to another method. ... 14일 전 L135  
Code Smell Blocker Open 활동되지 않음 30min effort 코멘트 cert, multi-threading, pitfall

→ 상속 받은 자식 클래스에서 super()를 호출했을 때 thread가 실행될 수 있다.

```
else if(str[1].equals("5")) str[1]="MAY";
```

Missing curly brace. ... 13일 전 L471  
Code Smell Critical Open 활동되지 않음 2min effort 코멘트 based-on-misra, cert, pitfall

At most one statement is allowed per line, but 2 statements were found on this line. ... 13일 전 L471  
Code Smell Major Open 활동되지 않음 1min effort 코멘트 style

→ 가급적이면 {} 을 사용하고, 하나의 line에 하나의 statement만 넣어야 한다.

```
try {  
    Thread.sleep(1000);  
}catch (InterruptedException e) {}
```

Either remove or fill this block of code. ... 12일 전 L27  
Code Smell Major Open 활동되지 않음 5min effort 코멘트 suspicious

→ 빈 code block은 채우거나 없애야 한다.

## 1.6 Coverage

\*SonarQube Coverage가 0%인 이슈가 있어 다음과 같이 IntelliJ의 Junit과 Eclipse 플러그인인 EclEmma를 활용하였다.

### 1.6.1 IntelliJ - Junit

우선 8조의 Unit Test code의 code coverage는 다음과 같다.

Element	Class, %	Method, %	Line, %
Alarm	100% (1/1)	100% (11/11)	94% (49/52)
CreateMode	100% (1/1)	100% (2/2)	57% (11/19)
DeleteMode	100% (1/1)	100% (2/2)	52% (13/25)
ModeSelector	100% (1/1)	75% (6/8)	85% (35/41)
Moonphase	100% (1/1)	40% (2/5)	25% (14/55)
Stopwatch	100% (1/1)	100% (15/15)	87% (51/58)
SystemUI	100% (6/6)	16% (8/49)	13% (87/667)
Tide	100% (1/1)	50% (3/6)	19% (31/157)
TimeDB	66% (2/3)	92% (13/14)	72% (54/75)
TimeKeeping	100% (1/1)	100% (5/5)	100% (17/17)
Timer	33% (1/3)	77% (14/18)	74% (50/67)

=> coverage %로 보았을 때, 7조보다 비교적 Unit Test code를 잘 작성하였음을 알 수 있다.

Unit Test를 실행시켜본 결과, 56개의 테스트 중에 55개를 통과하고, 단 1개를 실패하였다.

Tests failed: 1, passed: 55 of 56 tests – 4 s 469 ms

Package	Test Name	Duration
<default package>		4 s 469 ms
▶	AlarmTest	12 ms
▶	ModeSelectorTest	12 ms
▶	DeleteModeTest	6 ms
▶	TimeKeepingTest	21 ms
▶	MoonphaseTest	18 ms
▶	TimerTest	27 ms
▼	SystemUITest	4 s 354 ms
▶	run	0 ms
▶	main	4 s 354 ms
▶	TideTest	1 ms
▶	CreateModeTest	0 ms
▶	TimeDBTest	1 ms
▼	StopwatchTest	17 ms
▶	getInstance	0 ms
▶	getZeroSate	0 ms
▶	deleteInstance	1 ms
▶	getRunState	0 ms
▶	setStopwatch	1 ms
▶	pauseStopwatch	9 ms
▶	updateTime	0 ms
▶	getTime	0 ms
▶	recordStopwatch	0 ms
▶	startStopwatch	3 ms
▶	resetStopwatch	3 ms

실패한 테스트케이스는 MoonphaseTest에서 검출되었는데, 실패한 원인은 다음과 같다.

```

/Library/Java/JavaVirtualMachines/jdk1.8.0_201.jdk/Contents/Home/bin/java ...
— IntelliJ IDEA coverage runner —
sampling ...
include patterns:
exclude patterns:
java.lang.Exception: Method calculateMoonphase() should be public
  
```

### 1.5.2 Eclipse - EclEmma

\*전체 coverage와 테스트케이스가 지나가는 코드부분을 녹색, 지나가지 않는 부분을 적색으로 표시한 것을 보기 위해 Eclipse의 플러그인인 EclEmma를 활용하였다.

Element	Coverage ^	Covered Instructions	Missed Instructions	Total Instructions
digital-watch	75.5 %	13,880	4,510	18,390
src/main/java	74.6 %	13,180	4,480	17,660
(default package)	74.6 %	13,180	4,480	17,660
Moonphase.java	0.0 %	0	147	147
SystemUI.java	2.1 %	72	3,413	3,485
Alarm.java	53.2 %	107	94	201
TimeDB.java	63.7 %	225	128	353
DeleteMode.java	67.9 %	38	18	56
CreateMode.java	73.9 %	34	12	46
Timer.java	74.3 %	162	56	218
ModeSelector.java	87.5 %	126	18	144
Stopwatch.java	89.3 %	176	21	197
Tide.java	95.5 %	12,150	573	12,723
TimeKeeping.java	100.0 %	90	0	90

=> 7조와 비교해봤을 때, coverage %가 전체적으로 높다는 것을 알 수 있다.

CreateMode.java파일을 예로 들면, 다음과 같이 CreateModeTest의 테스트케이스가 지나가는 부분은 녹색, 지나가지 않는 부분은 적색으로 볼 수 있다.

(instruction의 일부만 통과된 경우 노란색이다)

```

20 private void createModeByList(){
21     for(int i=0;i<createlist.size();i++){
22         //일단 String값으로 비교를 하도록 설정했는데
23         switch (createlist.get(i)){
24             case "TimeKeeping":
25                 TimeKeeping.getInstance();
26                 break;
27             case "Timer":
28                 Timer.getInstance();
29                 break;
30             case "Alarm":
31                 Alarm.getInstance();
32                 break;
33             case "Stopwatch":
34                 Stopwatch.getInstance();
35                 break;
36             case "Tide":
37                 Tide.getInstance();
38                 break;
39             case "Moonphase":
40                 Moonphase.getInstance();
41                 break;
42             default:
43                 break;
44         }
45     }
46 }
47 }
  
```

## 1.7 Duplications

Overall	
밀도	3.6%
Duplicated lines	105
Duplicated blocks	10
Duplicated files	3

전체적으로 보았을 때, 중복 밀도는 3.6%이며, 중복된 라인 수는 105줄, 중복된 블록 수는 10개이다. 총 3개의 파일에서 중복이 검출되었다.

- 1) ModeSelector.java
- 2) Tide.java
- 3) SystemUI.java

Duplicated lines (%) 3.6% 

누수 기간: 이전 버전 이후

	Duplicated lines (%)	Duplicated lines
 digital-watch/src/main/java/ModeSelector.java	15.1%	18
 digital-watch/src/main/java/Tide.java	8.9%	45
 digital-watch/src/main/java/SystemUI.java	2.9%	42

## 1.8 Summary

- 마찬가지로 GUI 쪽 code에서 false alarm이 존재하였다.
- vulnerability의 경우 대부분이 접근 지정자 문제인 7조와 달리 copy store 문제로 인한 취약점이 추가로 존재하였다.
- 마찬가지로 code smell의 경우 초기화 문제와 같이 7조보다 좀 더 critical 하다고 판단되는 요소가 존재하였다.
- 7조에 비하여 unit test가 상당히 잘 진행되었으며 code coverage 또한 높았다. 이는 system testing의 pairwise test의 결과가 보여주듯 기능 부분쪽에서 Fail이 거의 없게 만들어주는 기반이 되었다고 볼 수 있다.
- 위에 상기한 문제를 수정하여 vulnerability와 code smell부분의 문제 수를 줄여나가야 한다.