



DVM System

3rd Cycle Development

Team 2

201810286 박혜린

201811264 변장훈

201311312 정인욱

INDEX

- 1 Redmine에 등록된 이슈
- 2 수정 진행과정
- 3 CTIP 환경에 대한 소감
- 4 UP 개발 방식에 대한 소감



01

Redmine에 등록된 이슈



01 Retesting

TestLink BFT, CPT 1차 분석, 2차 분석 비교 결과

Test Case ID	Pass/Fail
1차 Test	30 Pass, 24 Fail

1차 테스트
테스트를 100% 진행했을 경우 89가지 테스트케이스에 대하여 69.66% Pass, 30.34% Fail

2차 테스트
테스트를 100% 진행했을 경우 89가지 테스트케이스에 대하여 100% Pass

Test Case ID	Pass/Fail
2차 Test	54 Pass, 0 Fail

Category Partition Testing Summary

	Total(개)	Pass(개)	Fail(개)	Pass(%)
1차 Test	54	30	24	55.6
2차 Test	54	54	0	100

CPT Result : 54/54 = 100 % passed

Brute-force Testing Summary

	Total(개)	Pass(개)	Fail(개)	Pass(%)
1차 Test	35	32	3	91
2차 Test	35	35	0	100

BFT Result : 35/35 = 100 % passed



Redmine에 등록된 이슈

1. `math.random()` 보안 취약 (#39)
2. `printStackTrace()` 보안 취약 (#40~48)
3. Null pointer 참조 (#49)
4. Resource closed 문제 (파일 입출력) (#50~53)
5. Resource closed 문제 (소켓 관련) (#54~57, #59)
6. 주석 처리된 코드 (#58, #60~65)
7. 정적 변수 관련 문제 (#66~67)
8. 코드 중복 (#69)
9. 기타 문제 (#68, #73)
10. GUI 함수들 관련 문제 (#70, #72, #74)



1. math.random() 보안 취약

39 결함 신규 보통 Security Hotspot-1 : Math.random() 2021/05/28 04:49 ...

Security Hotspot-1 : Math.random() < 뒤로 | 35/35 | 다음 >

admin Admin이(가) 11일 전에 추가함.

상태:	신규	시작시간:	2021/05/28
우선순위:	보통	완료기한:	
담당자:	-		
Testing:	Security Hotspots		

설명 댓글달기

위치 : VerificationCodeList.java / Line:37 / Math.random()

java.lang.Math.random() method는 보안이 취약합니다. java.security.SecureRandom RNG 사용을 추천합니다.

하위 일감 추가

연결된 일감 추가



Redmine에 등록된 이슈

2. printStackTrace() 보안 취약

<input type="checkbox"/>	48	결함	해결	낮음	Security Hotspot-10 : printStackTrace()	2021/06/08 00:20	...
<input type="checkbox"/>	47	결함	해결	낮음	Security Hotspot-9 : printStackTrace()	2021/06/08 00:20	...
<input type="checkbox"/>	46	결함	해결	낮음	Security Hotspot-8 : printStackTrace()	2021/06/08 00:20	...
<input type="checkbox"/>	45	결함	해결	낮음	Security Hotspot-7 : printStackTrace()	2021/06/08 00:20	...
<input type="checkbox"/>	44	결함	해결	낮음	Security Hotspot-6 : printStackTrace()	2021/06/08 00:19	...
<input type="checkbox"/>	43	결함	해결	낮음	Security Hotspot-5 : printStackTrace()	2021/06/08 00:19	...
<input type="checkbox"/>	42	결함	해결	낮음	Security Hotspot-4 : printStackTrace()	2021/06/08 00:19	...
<input type="checkbox"/>	41	결함	해결	낮음	Security Hotspot-3 : printStackTrace()	2021/06/08 00:19	...
<input type="checkbox"/>	40	결함	해결	낮음	Security Hotspot-2 : printStackTrace()	2021/06/08 00:19	...

Security Hotspot-2 : printStackTrace() « 뒤로 | 34/35 | 다음 »

admin Admin이(가) 11일 전에 추가함. 약 10시간 전에 수정됨.

상태:	해결	시작시간:	2021/05/28
우선순위:	낮음	완료기한:	
담당자:	-		
Testing:	Security Hotspots		

설명 💬 댓글달기

위치 : CardList.java / Line 52

printStackTrace()는 공격자의 공격을 용이하게 합니다.

하위 일감 추가

연결된 일감 추가



2. printStackTrace() 보안 취약

<input type="checkbox"/>	48	결함	해결	낮음	Security Hotspot-10 : printStackTrace()	2021/06/08 00:20	...
<input type="checkbox"/>	47	결함	해결	낮음	Security Hotspot-9 : printStackTrace()	2021/06/08 00:20	...
<input type="checkbox"/>	46	결함	해결	낮음	Security Hotspot-8 : printStackTrace()	2021/06/08 00:20	...
<input type="checkbox"/>	45	결함	해결	낮음	Security Hotspot-7 : printStackTrace()	2021/06/08 00:20	...
<input type="checkbox"/>	44	결함	해결	낮음	Security Hotspot-6 : printStackTrace()	2021/06/08 00:19	...
<input type="checkbox"/>	43	결함	해결	낮음	Security Hotspot-5 : printStackTrace()	2021/06/08 00:19	...
<input type="checkbox"/>	42	결함	해결	낮음	Security Hotspot-4 : printStackTrace()	2021/06/08 00:19	...
<input type="checkbox"/>	41	결함	해결	낮음	Security Hotspot-3 : printStackTrace()	2021/06/08 00:19	...
<input type="checkbox"/>	40	결함	해결	낮음	Security Hotspot-2 : printStackTrace()	2021/06/08 00:19	...

Security Hotspot-3 : printStackTrace()

« 뒤로 | 33/35 | 다음 »

admin Admin이(가) 11일 전에 추가함. 약 10시간 전에 수정됨.

상태:	해결	시작시간:	2021/05/28
우선순위:	낮음	완료기한:	
담당자:	-		
Testing:	Security Hotspots		

설명

위치 : Controller.java / Line 71

printStackTrace()는 공격자의 공격을 용이하게 합니다.



2. printStackTrace() 보안 취약

<input type="checkbox"/>	48	결함	해결	낮음	Security Hotspot-10 : printStackTrace()	2021/06/08 00:20	...
<input type="checkbox"/>	47	결함	해결	낮음	Security Hotspot-9 : printStackTrace()	2021/06/08 00:20	...
<input type="checkbox"/>	46	결함	해결	낮음	Security Hotspot-8 : printStackTrace()	2021/06/08 00:20	...
<input type="checkbox"/>	45	결함	해결	낮음	Security Hotspot-7 : printStackTrace()	2021/06/08 00:20	...
<input type="checkbox"/>	44	결함	해결	낮음	Security Hotspot-6 : printStackTrace()	2021/06/08 00:19	...
<input type="checkbox"/>	43	결함	해결	낮음	Security Hotspot-5 : printStackTrace()	2021/06/08 00:19	...
<input type="checkbox"/>	42	결함	해결	낮음	Security Hotspot-4 : printStackTrace()	2021/06/08 00:19	...
<input type="checkbox"/>	41	결함	해결	낮음	Security Hotspot-3 : printStackTrace()	2021/06/08 00:19	...
<input type="checkbox"/>	40	결함	해결	낮음	Security Hotspot-2 : printStackTrace()	2021/06/08 00:19	...

Security Hotspot-4 : printStackTrace()

admin Admin이(가) 11일 전에 추가함. 약 10시간 전에 수정됨.

상태:	해결	시작시간:	2021/05/28
우선순위:	낮음	완료기한:	
담당자:	-		
Testing:	Security Hotspots		

설명

위치 : Controller.java / Line 84

printStackTrace()는 공격자의 공격을 용이하게 합니다.



Redmine에 등록된 이슈

2. printStackTrace() 보안 취약

<input type="checkbox"/>	48	결함	해결	낮음	Security Hotspot-10 : printStackTrace()	2021/06/08 00:20	...
<input type="checkbox"/>	47	결함	해결	낮음	Security Hotspot-9 : printStackTrace()	2021/06/08 00:20	...
<input type="checkbox"/>	46	결함	해결	낮음	Security Hotspot-8 : printStackTrace()	2021/06/08 00:20	...
<input type="checkbox"/>	45	결함	해결	낮음	Security Hotspot-7 : printStackTrace()	2021/06/08 00:20	...
<input type="checkbox"/>	44	결함	해결	낮음	Security Hotspot-6 : printStackTrace()	2021/06/08 00:19	...
<input type="checkbox"/>	43	결함	해결	낮음	Security Hotspot-5 : printStackTrace()	2021/06/08 00:19	...
<input type="checkbox"/>	42	결함	해결	낮음	Security Hotspot-4 : printStackTrace()	2021/06/08 00:19	...
<input type="checkbox"/>	41	결함	해결	낮음	Security Hotspot-3 : printStackTrace()	2021/06/08 00:19	...
<input type="checkbox"/>	40	결함	해결	낮음	Security Hotspot-2 : printStackTrace()	2021/06/08 00:19	...

Security Hotspot-5 : printStackTrace()

admin Admin이(가) 11일 전에 추가함. 약 10시간 전에 수정됨.

상태:	해결	시작시간:	2021/05/28
우선순위:	낮음	완료기한:	
담당자:	-		
Testing:	Security Hotspots		

설명

위치 : Controller.java / Line 100

printStackTrace()는 공격자의 공격을 용이하게 합니다.



2. printStackTrace() 보안 취약

<input type="checkbox"/>	48	결함	해결	낮음	Security Hotspot-10 : printStackTrace()	2021/06/08 00:20	...
<input type="checkbox"/>	47	결함	해결	낮음	Security Hotspot-9 : printStackTrace()	2021/06/08 00:20	...
<input type="checkbox"/>	46	결함	해결	낮음	Security Hotspot-8 : printStackTrace()	2021/06/08 00:20	...
<input type="checkbox"/>	45	결함	해결	낮음	Security Hotspot-7 : printStackTrace()	2021/06/08 00:20	...
<input type="checkbox"/>	44	결함	해결	낮음	Security Hotspot-6 : printStackTrace()	2021/06/08 00:19	...
<input type="checkbox"/>	43	결함	해결	낮음	Security Hotspot-5 : printStackTrace()	2021/06/08 00:19	...
<input type="checkbox"/>	42	결함	해결	낮음	Security Hotspot-4 : printStackTrace()	2021/06/08 00:19	...
<input type="checkbox"/>	41	결함	해결	낮음	Security Hotspot-3 : printStackTrace()	2021/06/08 00:19	...
<input type="checkbox"/>	40	결함	해결	낮음	Security Hotspot-2 : printStackTrace()	2021/06/08 00:19	...

Security Hotspot-6 : printStackTrace()

admin Admin이(가) 11일 전에 추가함. 약 10시간 전에 수정됨.

상태:	해결	시작시간:	2021/05/28
우선순위:	낮음	완료기한:	
담당자:	-		
Testing:	Security Hotspots		

설명

위치 : Controller.java / Line 227

printStackTrace()는 공격자의 공격을 용이하게 합니다.



Redmine에 등록된 이슈

2. printStackTrace() 보안 취약

<input type="checkbox"/>	48	결함	해결	낮음	Security Hotspot-10 : printStackTrace()	2021/06/08 00:20	...
<input type="checkbox"/>	47	결함	해결	낮음	Security Hotspot-9 : printStackTrace()	2021/06/08 00:20	...
<input type="checkbox"/>	46	결함	해결	낮음	Security Hotspot-8 : printStackTrace()	2021/06/08 00:20	...
<input type="checkbox"/>	45	결함	해결	낮음	Security Hotspot-7 : printStackTrace()	2021/06/08 00:20	...
<input type="checkbox"/>	44	결함	해결	낮음	Security Hotspot-6 : printStackTrace()	2021/06/08 00:19	...
<input type="checkbox"/>	43	결함	해결	낮음	Security Hotspot-5 : printStackTrace()	2021/06/08 00:19	...
<input type="checkbox"/>	42	결함	해결	낮음	Security Hotspot-4 : printStackTrace()	2021/06/08 00:19	...
<input type="checkbox"/>	41	결함	해결	낮음	Security Hotspot-3 : printStackTrace()	2021/06/08 00:19	...
<input type="checkbox"/>	40	결함	해결	낮음	Security Hotspot-2 : printStackTrace()	2021/06/08 00:19	...

Security Hotspot-7 : printStackTrace()

admin Admin이(가) 11일 전에 추가함. 약 10시간 전에 수정됨.

상태:	해결	시작시간:	2021/05/28
우선순위:	낮음	완료기한:	
담당자:	-		
Testing:	Security Hotspots		

설명

위치 : Network.java / Line 102

printStackTrace()는 공격자의 공격을 용이하게 합니다.



2. printStackTrace() 보안 취약

<input type="checkbox"/>	48	결함	해결	낮음	Security Hotspot-10 : printStackTrace()	2021/06/08 00:20	...
<input type="checkbox"/>	47	결함	해결	낮음	Security Hotspot-9 : printStackTrace()	2021/06/08 00:20	...
<input type="checkbox"/>	46	결함	해결	낮음	Security Hotspot-8 : printStackTrace()	2021/06/08 00:20	...
<input type="checkbox"/>	45	결함	해결	낮음	Security Hotspot-7 : printStackTrace()	2021/06/08 00:20	...
<input type="checkbox"/>	44	결함	해결	낮음	Security Hotspot-6 : printStackTrace()	2021/06/08 00:19	...
<input type="checkbox"/>	43	결함	해결	낮음	Security Hotspot-5 : printStackTrace()	2021/06/08 00:19	...
<input type="checkbox"/>	42	결함	해결	낮음	Security Hotspot-4 : printStackTrace()	2021/06/08 00:19	...
<input type="checkbox"/>	41	결함	해결	낮음	Security Hotspot-3 : printStackTrace()	2021/06/08 00:19	...
<input type="checkbox"/>	40	결함	해결	낮음	Security Hotspot-2 : printStackTrace()	2021/06/08 00:19	...

Security Hotspot-8 : printStackTrace()

admin Admin이(가) 11일 전에 추가함. 약 10시간 전에 수정됨.

상태:	해결	시작시간:	2021/05/28
우선순위:	낮음	완료기한:	
담당자:	-		
Testing:	Security Hotspots		

설명

위치 : Network.java / Line 151

printStackTrace()는 공격자의 공격을 용이하게 합니다.



2. printStackTrace() 보안 취약

<input type="checkbox"/>	48	결함	해결	낮음	Security Hotspot-10 : printStackTrace()	2021/06/08 00:20	...
<input type="checkbox"/>	47	결함	해결	낮음	Security Hotspot-9 : printStackTrace()	2021/06/08 00:20	...
<input type="checkbox"/>	46	결함	해결	낮음	Security Hotspot-8 : printStackTrace()	2021/06/08 00:20	...
<input type="checkbox"/>	45	결함	해결	낮음	Security Hotspot-7 : printStackTrace()	2021/06/08 00:20	...
<input type="checkbox"/>	44	결함	해결	낮음	Security Hotspot-6 : printStackTrace()	2021/06/08 00:19	...
<input type="checkbox"/>	43	결함	해결	낮음	Security Hotspot-5 : printStackTrace()	2021/06/08 00:19	...
<input type="checkbox"/>	42	결함	해결	낮음	Security Hotspot-4 : printStackTrace()	2021/06/08 00:19	...
<input type="checkbox"/>	41	결함	해결	낮음	Security Hotspot-3 : printStackTrace()	2021/06/08 00:19	...
<input type="checkbox"/>	40	결함	해결	낮음	Security Hotspot-2 : printStackTrace()	2021/06/08 00:19	...

Security Hotspot-9 : printStackTrace()

admin Admin이(가) 11일 전에 추가함. 약 10시간 전에 수정됨.

상태:	해결	시작시간:	2021/05/28
우선순위:	낮음	완료기한:	
담당자:	-		
Testing:	Security Hotspots		

설명

위치 : Network.java / Line 196

printStackTrace()는 공격자의 공격을 용이하게 합니다.



2. printStackTrace() 보안 취약

<input type="checkbox"/>	48	결함	해결	낮음	Security Hotspot-10 : printStackTrace()	2021/06/08 00:20	...
<input type="checkbox"/>	47	결함	해결	낮음	Security Hotspot-9 : printStackTrace()	2021/06/08 00:20	...
<input type="checkbox"/>	46	결함	해결	낮음	Security Hotspot-8 : printStackTrace()	2021/06/08 00:20	...
<input type="checkbox"/>	45	결함	해결	낮음	Security Hotspot-7 : printStackTrace()	2021/06/08 00:20	...
<input type="checkbox"/>	44	결함	해결	낮음	Security Hotspot-6 : printStackTrace()	2021/06/08 00:19	...
<input type="checkbox"/>	43	결함	해결	낮음	Security Hotspot-5 : printStackTrace()	2021/06/08 00:19	...
<input type="checkbox"/>	42	결함	해결	낮음	Security Hotspot-4 : printStackTrace()	2021/06/08 00:19	...
<input type="checkbox"/>	41	결함	해결	낮음	Security Hotspot-3 : printStackTrace()	2021/06/08 00:19	...
<input type="checkbox"/>	40	결함	해결	낮음	Security Hotspot-2 : printStackTrace()	2021/06/08 00:19	...

Security Hotspot-10 : printStackTrace()

admin Admin이(가) 11일 전에 추가함. 약 10시간 전에 수정됨.

상태:	해결	시작시간:	2021/05/28
우선순위:	낮음	완료기한:	
담당자:	-		
Testing:	Security Hotspots		

설명

위치 : Network.java / Line 224

printStackTrace()는 공격자의 공격을 용이하게 합니다.



3. Null pointer 참조

49 결함 해결 보통 Bug-1 : Null pointers should not be dereferenced 2021/06/08 00:15 ...

Bug-1 : Null pointers should not be dereferenced

admin Admin이(가) 11일 전에 추가함. 약 10시간 전에 수정됨.

상태:	해결	시작시간:	2021/05/28
우선순위:	보통	완료기한:	
담당자:	-		
Testing:	Bugs		

설명

위치 : Network.java / Line : 96

Line 25 에서 Msg sendMsg = null 이라고 선언하고 Line 96에서 sendMsg.getSrc_id()를 하는 것은 NullPointerException 예러가 발생할 가능성이 있다.

따라서 NullPointerException 처리가 필요하다.

4. Resource closed 문제 (파일 입출력)

<input type="checkbox"/>	53	결함	해결	긴급	Bug-5 : Resources should be closed	2021/06/08 00:28	...
<input type="checkbox"/>	52	결함	해결	긴급	Bug-4 : Resources should be closed	2021/06/08 00:27	...
<input type="checkbox"/>	51	결함	해결	긴급	Bug-3 : Resources should be closed	2021/06/08 00:26	...
<input type="checkbox"/>	50	결함	해결	긴급	Bug-2 : Resources should be closed	2021/06/08 00:23	...

Bug-2 : Resources should be closed

admin Admin이(가) 11일 전에 추가함. 약 10시간 전에 수정됨.

상태:	해결	시작시간:	2021/05/28
우선순위:	긴급	완료기한:	
담당자:	-		
Testing:	Bugs		

설명

위치 : CardList.java / Line : 28

위 리소스는 Closeable 해야 합니다. 닫지 않는다면 예외로 인해 호출이 수행되지 않을 수 있습니다.

AutoCloseable 클래스 'try-with-resources'패턴을 사용하는 것을 추천합니다.

4. Resource closed 문제 (파일 입출력)

<input type="checkbox"/>	53	결함	해결	긴급	Bug-5 : Resources should be closed	2021/06/08 00:28	...
<input type="checkbox"/>	52	결함	해결	긴급	Bug-4 : Resources should be closed	2021/06/08 00:27	...
<input type="checkbox"/>	51	결함	해결	긴급	Bug-3 : Resources should be closed	2021/06/08 00:26	...
<input type="checkbox"/>	50	결함	해결	긴급	Bug-2 : Resources should be closed	2021/06/08 00:23	...

Bug-3 : Resources should be closed

admin Admin이(가) 11일 전에 추가함. 약 10시간 전에 수정됨.

상태:	해결	시작시간:	2021/05/28
우선순위:	긴급	완료기한:	
담당자:	-		
Testing:	Bugs		

설명

위치 : Controller.java / Line : 38

위 리소스는 Closeable 해야 합니다. 닫지 않는다면 예외로 인해 호출이 수행되지 않을 수 있습니다.

AutoCloseable 클래스 'try-with-resources'패턴을 사용하는 것을 추천합니다.

예시 :

```
private void readTheFile() throws IOException {
    Path path = Paths.get(this.fileName);
    BufferedReader reader = Files.newBufferedReader(path, this.charset);
    // ...
    reader.close(); // Noncompliant
    // ...
    Files.lines("input.txt").forEach(System.out::println); // Noncompliant: The stream needs to be closed
}
```

```
private void doSomething() {
    OutputStream stream = null;
    try {
        for (String property : propertyList) {
            stream = new FileOutputStream("myfile.txt"); // Noncompliant
            // ...
        }
    } catch (Exception e) {
        // ...
    } finally {
        stream.close(); // Multiple streams were opened. Only the last is closed.
    }
}
```

4. Resource closed 문제 (파일 입출력)

<input type="checkbox"/>	53	결함	해결	긴급	Bug-5 : Resources should be closed	2021/06/08 00:28	...
<input type="checkbox"/>	52	결함	해결	긴급	Bug-4 : Resources should be closed	2021/06/08 00:27	...
<input type="checkbox"/>	51	결함	해결	긴급	Bug-3 : Resources should be closed	2021/06/08 00:26	...
<input type="checkbox"/>	50	결함	해결	긴급	Bug-2 : Resources should be closed	2021/06/08 00:23	...

Bug-4 : Resources should be closed

admin Admin이(가) 11일 전에 추가함. 약 10시간 전에 수정됨.

상태:	해결	시작시간:	2021/05/28
우선순위:	긴급	완료기한:	
담당자:	-		
Testing:	Bugs		

설명

위치 : Controller.java / Line : 89

위 리소스는 Closeable 해야 합니다. 닫지 않는다면 예외로 인해 호출이 수행되지 않을 수 있습니다.

AutoCloseable 클래스 'try-with-resources'패턴을 사용하는 것을 추천합니다.

```

예시 :
private void readTheFile() throws IOException {
    Path path = Paths.get(this.fileName);
    BufferedReader reader = Files.newBufferedReader(path, this.charset);
    // ...
    reader.close(); // Noncompliant
    // ...
    Files.lines("input.txt").forEach(System.out::println); // Noncompliant: The stream needs to be closed
}

private void doSomething() {
    OutputStream stream = null;
    try {
        for (String property : propertyList) {
            stream = new FileOutputStream("myfile.txt"); // Noncompliant
            // ...
        }
    } catch (Exception e) {
        // ...
    } finally {
        stream.close(); // Multiple streams were opened. Only the last is closed.
    }
}

```

4. Resource closed 문제 (파일 입출력)

<input type="checkbox"/>	53	결함	해결	긴급	Bug-5 : Resources should be closed	2021/06/08 00:28	...
<input type="checkbox"/>	52	결함	해결	긴급	Bug-4 : Resources should be closed	2021/06/08 00:27	...
<input type="checkbox"/>	51	결함	해결	긴급	Bug-3 : Resources should be closed	2021/06/08 00:26	...
<input type="checkbox"/>	50	결함	해결	긴급	Bug-2 : Resources should be closed	2021/06/08 00:23	...

Bug-5 : Resources should be closed

admin Admin이(가) 11일 전에 추가함. 약 10시간 전에 수정됨.

상태:	해결	시작시간:	2021/05/28
우선순위:	긴급	완료기한:	
담당자:	-		
Testing:	Bugs		

설명

위치 : Controller.java / Line : 210

위 리소스는 Closeable 해야 합니다. 닫지 않으면 예외로 인해 호출이 수행되지 않을 수 있습니다.

AutoCloseable 클래스 'try-with-resources' 패턴을 사용하는 것을 추천합니다.

```

예시 :
private void readTheFile() throws IOException {
    Path path = Paths.get(this.fileName);
    BufferedReader reader = Files.newBufferedReader(path, this.charset);
    // ...
    reader.close(); // Noncompliant
    // ...
    Files.lines("input.txt").forEach(System.out::println); // Noncompliant: The stream needs to be closed
}

private void doSomething() {
    OutputStream stream = null;
    try {
        for (String property : propertyList) {
            stream = new FileOutputStream("myfile.txt"); // Noncompliant
            // ...
        }
    } catch (Exception e) {
        // ...
    } finally {
        stream.close(); // Multiple streams were opened. Only the last is closed.
    }
}

```



5. Resource closed 문제 (소켓 관련)

<input type="checkbox"/>	57	결함	신규	긴급	Bug-9 : Resources should be closed	2021/05/28 05:14	...
<input type="checkbox"/>	56	결함	신규	긴급	Bug-8 : Resources should be closed	2021/05/28 05:13	...
<input type="checkbox"/>	55	결함	신규	긴급	Bug-7 : Resources should be closed	2021/05/28 05:13	...
<input type="checkbox"/>	54	결함	신규	긴급	Bug-6 : Resources should be closed	2021/05/28 05:13	...
<input type="checkbox"/>	59	결함	신규	긴급	Bug-10 : Resources should be closed	2021/05/28 05:16	...

Bug-6 : Resources should be closed
 admin Admin이(가) 11일 전에 추가함.

상태:	신규	시작시간:	2021/05/28
우선순위:	긴급	완료기한:	
담당자:	-		
Testing:	Bugs		

설명
 위치 : Network.java / Line : 31
 위 리소스는 Closeable 해야 합니다. 닫지 않는다면 예외로 인해 호출이 수행되지 않을 수 있습니다.
 AutoCloseable 클래스 'try-with-resources' 패턴을 사용하는 것을 추천합니다.

예시 :

```
private void readTheFile() throws IOException {
    Path path = Paths.get(this.fileName);
    BufferedReader reader = Files.newBufferedReader(path, this.charset);
    // ...
    reader.close(); // Noncompliant
    // ...
    Files.lines("input.txt").forEach(System.out::println); // Noncompliant: The stream needs to be closed
}

private void doSomething() {
    OutputStream stream = null;
    try {
        for (String property : propertyList) {
            stream = new FileOutputStream("myfile.txt"); // Noncompliant
            // ...
        }
        catch (Exception e) {
            // ...
        }
        finally {
            stream.close(); // Multiple streams were opened. Only the last is closed.
        }
    }
}
```

5. Resource closed 문제 (소켓 관련)

<input type="checkbox"/>	57	결함	신규	긴급	Bug-9 : Resources should be closed	2021/05/28 05:14	...
<input type="checkbox"/>	56	결함	신규	긴급	Bug-8 : Resources should be closed	2021/05/28 05:13	...
<input type="checkbox"/>	55	결함	신규	긴급	Bug-7 : Resources should be closed	2021/05/28 05:13	...
<input type="checkbox"/>	54	결함	신규	긴급	Bug-6 : Resources should be closed	2021/05/28 05:13	...
<input type="checkbox"/>	59	결함	신규	긴급	Bug-10 : Resources should be closed	2021/05/28 05:16	...

Bug-7 : Resources should be closed

admin Admin이(가) 11일 전에 추가함.

상태: 신규
 우선순위: 긴급
 담당자: -
 Testing: Bugs

시작시간: 2021/05/28
 완료기한:

설명

위치 : Network.java / Line : 112

위 리소스는 Closeable 해야 합니다. 닫지 않는다면 예외로 인해 호출이 수행되지 않을 수 있습니다.

AutoCloseable 클래스 'try-with-resources' 패턴을 사용하는 것을 추천합니다.

```

예시 :
private void readTheFile() throws IOException {
    Path path = Paths.get(this.fileName);
    BufferedReader reader = Files.newBufferedReader(path, this.charset);
    // ...
    reader.close(); // Noncompliant
    // ...
    Files.lines("input.txt").forEach(System.out::println); // Noncompliant: The stream needs to be closed
}

private void doSomething() {
    OutputStream stream = null;
    try {
        for (String property : propertyList) {
            stream = new FileOutputStream("myfile.txt"); // Noncompliant
            // ...
        }
    } catch (Exception e) {
        // ...
    } finally {
        stream.close(); // Multiple streams were opened. Only the last is closed.
    }
}

```



5. Resource closed 문제 (소켓 관련)

<input type="checkbox"/>	57	결함	신규	긴급	Bug-9 : Resources should be closed	2021/05/28 05:14	...
<input type="checkbox"/>	56	결함	신규	긴급	Bug-8 : Resources should be closed	2021/05/28 05:13	...
<input type="checkbox"/>	55	결함	신규	긴급	Bug-7 : Resources should be closed	2021/05/28 05:13	...
<input type="checkbox"/>	54	결함	신규	긴급	Bug-6 : Resources should be closed	2021/05/28 05:13	...
<input type="checkbox"/>	59	결함	신규	긴급	Bug-10 : Resources should be closed	2021/05/28 05:16	...

Bug-8 : Resources should be closed

admin Admin이(가) 11일 전에 추가함.

상태: 신규
 우선순위: 긴급
 담당자: -
 Testing: Bugs

시작시간: 2021/05/28
 완료기한:

설명

위치 : Network.java / Line : 141

위 리소스는 Closeable 해야 합니다. 닫지 않으면 예외로 인해 호출이 수행되지 않을 수 있습니다.

AutoCloseable 클래스 'try-with-resources' 패턴을 사용하는 것을 추천합니다.

```

예시 :
private void readTheFile() throws IOException {
    Path path = Paths.get(this.fileName);
    BufferedReader reader = Files.newBufferedReader(path, this.charset);
    // ...
    reader.close(); // Noncompliant
    // ...
    Files.lines("input.txt").forEach(System.out::println); // Noncompliant: The stream needs to be closed
}

private void doSomething() {
    OutputStream stream = null;
    try {
        for (String property : propertyList) {
            stream = new FileOutputStream("myfile.txt"); // Noncompliant
            // ...
        }
    } catch (Exception e) {
        // ...
    } finally {
        stream.close(); // Multiple streams were opened. Only the last is closed.
    }
}

```



5. Resource closed 문제 (소켓 관련)

<input type="checkbox"/>	57	결함	신규	긴급	Bug-9 : Resources should be closed	2021/05/28 05:14	...
<input type="checkbox"/>	56	결함	신규	긴급	Bug-8 : Resources should be closed	2021/05/28 05:13	...
<input type="checkbox"/>	55	결함	신규	긴급	Bug-7 : Resources should be closed	2021/05/28 05:13	...
<input type="checkbox"/>	54	결함	신규	긴급	Bug-6 : Resources should be closed	2021/05/28 05:13	...
<input type="checkbox"/>	59	결함	신규	긴급	Bug-10 : Resources should be closed	2021/05/28 05:16	...

Bug-9 : Resources should be closed
admin Admin이(가) 11일 전에 추가함.

상태:	신규	시작시간:	2021/05/28
우선순위:	긴급	완료기한:	
담당자:	-		
Testing:	Bugs		

설명
위치 : Network.java / Line : 178
위 리소스는 Closeable 해야 합니다. 닫지 않는다면 예외로 인해 호출이 수행되지 않을 수 있습니다.
AutoCloseable 클래스 'try-with-resources' 패턴을 사용하는 것을 추천합니다.

예시 :

```
private void readTheFile() throws IOException {
    Path path = Paths.get(this.fileName);
    BufferedReader reader = Files.newBufferedReader(path, this.charset);
    // ...
    reader.close(); // Noncompliant
    // ...
    Files.lines("input.txt").forEach(System.out::println); // Noncompliant: The stream needs to be closed
}

private void doSomething() {
    OutputStream stream = null;
    try {
        for (String property : propertyList) {
            stream = new FileOutputStream("myfile.txt"); // Noncompliant
            // ...
        }
    } catch (Exception e) {
        // ...
    } finally {
        stream.close(); // Multiple streams were opened. Only the last is closed.
    }
}
```




5. Resource closed 문제 (소켓 관련)

<input type="checkbox"/>	57	결함	신규	긴급	Bug-9 : Resources should be closed	2021/05/28 05:14	...
<input type="checkbox"/>	56	결함	신규	긴급	Bug-8 : Resources should be closed	2021/05/28 05:13	...
<input type="checkbox"/>	55	결함	신규	긴급	Bug-7 : Resources should be closed	2021/05/28 05:13	...
<input type="checkbox"/>	54	결함	신규	긴급	Bug-6 : Resources should be closed	2021/05/28 05:13	...
<input type="checkbox"/>	59	결함	신규	긴급	Bug-10 : Resources should be closed	2021/05/28 05:16	...

Bug-10 : Resources should be closed

admin Admin이(가) 11일 전에 추가함.

상태:	신규	시작시간:	2021/05/28
우선순위:	긴급	완료기한:	
담당자:	-		
Testing:	Bugs		

설명

위치 : Network.java / Line : 208

위 리소스는 Closeable 해야 합니다. 닫지 않는다면 예외로 인해 호출이 수행되지 않을 수 있습니다.

AutoCloseable 클래스 'try-with-resources' 패턴을 사용하는 것 또는 'Socket'을 'finally'안에 넣는 것을 추천합니다.

```

예시 :
private void readTheFile() throws IOException {
    Path path = Paths.get(this.fileName);
    BufferedReader reader = Files.newBufferedReader(path, this.charset);
    // ...
    reader.close(); // Noncompliant
    // ...
    Files.lines("input.txt").forEach(System.out::println); // Noncompliant: The stream needs to be closed
}

private void doSomething() {
    OutputStream stream = null;
    try {
        for (String property : propertyList) {
            stream = new FileOutputStream("myfile.txt"); // Noncompliant
        }
    } catch (Exception e) {
    }
    // ...
} finally {
    stream.close(); // Multiple streams were opened. Only the last is closed.
}

```



6. 주석 처리된 코드

58 결함 해결 낮음 [Major]: This block of commented-out lines of code should be removed. - Network.java 2021/06/08 00:45 ...

<input type="checkbox"/>	65	결함	해결	낮음	[Major]: This block of commented-out lines of code should be removed. - VMList.java	2021/06/08 00:49	...
<input type="checkbox"/>	64	결함	해결	낮음	[Major]: This block of commented-out lines of code should be removed. - VM.java	2021/06/08 00:48	...
<input type="checkbox"/>	63	결함	해결	낮음	[Major]: This block of commented-out lines of code should be removed. - Card.java	2021/06/08 00:47	...
<input type="checkbox"/>	62	결함	해결	낮음	[Major]: This block of commented-out lines of code should be removed. - GUI_VMList.java	2021/06/08 00:47	...
<input type="checkbox"/>	61	결함	해결	낮음	[Major]: This block of commented-out lines of code should be removed. - GUI_Frame.java	2021/06/08 00:47	...
<input type="checkbox"/>	60	결함	해결	낮음	[Major]: This block of commented-out lines of code should be removed. - Controller.java	2021/06/08 00:46	...

[Major]: This block of commented-out lines of code should be removed. - Network.java

[admin Admin](#)이(가) 11일 전에 추가함. 약 10시간 전에 수정됨.

상태:	해결	시작시간:	2021/05/28
우선순위:	낮음	완료기한:	
담당자:	-		
Testing:	Code Smells		

설명
 위치: Network.java / Line: 239-284 , 127
 코드의 가독성을 높이기 위해서는 사용하지 않는 함수를 주석처리 대신 삭제해주는 것이 좋다.



6. 주석 처리된 코드

58 결함 해결 낮음 [Major]: This block of commented-out lines of code should be removed. - Network.java 2021/06/08 00:45 ...

<input type="checkbox"/>	65	결함	해결	낮음	[Major]: This block of commented-out lines of code should be removed. - VMList.java	2021/06/08 00:49	...
<input type="checkbox"/>	64	결함	해결	낮음	[Major]: This block of commented-out lines of code should be removed. - VM.java	2021/06/08 00:48	...
<input type="checkbox"/>	63	결함	해결	낮음	[Major]: This block of commented-out lines of code should be removed. - Card.java	2021/06/08 00:47	...
<input type="checkbox"/>	62	결함	해결	낮음	[Major]: This block of commented-out lines of code should be removed. - GUI_VMList.java	2021/06/08 00:47	...
<input type="checkbox"/>	61	결함	해결	낮음	[Major]: This block of commented-out lines of code should be removed. - GUI_Frame.java	2021/06/08 00:47	...
<input type="checkbox"/>	60	결함	해결	낮음	[Major]: This block of commented-out lines of code should be removed. - Controller.java	2021/06/08 00:46	...

[Major]: This block of commented-out lines of code should be removed. - Controller.java

admin Admin이(가) 11일 전에 추가함. 약 10시간 전에 수정됨.

상태:	해결	시작시간:	2021/05/28
우선순위:	낮음	완료기한:	
담당자:	-		
Testing:	Code Smells		

설명
 위치 : Controller.java / Line : 252
 코드의 가독성을 높이기 위해서는 사용하지 않는 함수를 주석처리 대신 삭제해주는 것이 좋다.



6. 주석 처리된 코드

<input type="checkbox"/>	58	결함	해결	낮음	[Major]: This block of commented-out lines of code should be removed. - Network.java	2021/06/08 00:45	...
<input type="checkbox"/>	65	결함	해결	낮음	[Major]: This block of commented-out lines of code should be removed. - VMList.java	2021/06/08 00:49	...
<input type="checkbox"/>	64	결함	해결	낮음	[Major]: This block of commented-out lines of code should be removed. - VM.java	2021/06/08 00:48	...
<input type="checkbox"/>	63	결함	해결	낮음	[Major]: This block of commented-out lines of code should be removed. - Card.java	2021/06/08 00:47	...
<input type="checkbox"/>	62	결함	해결	낮음	[Major]: This block of commented-out lines of code should be removed. - GUI_VMList.java	2021/06/08 00:47	...
<input type="checkbox"/>	61	결함	해결	낮음	[Major]: This block of commented-out lines of code should be removed. - GUI_Frame.java	2021/06/08 00:47	...
<input type="checkbox"/>	60	결함	해결	낮음	[Major]: This block of commented-out lines of code should be removed. - Controller.java	2021/06/08 00:46	...

[Major]: This block of commented-out lines of code should be removed. - GUI_Frame.java

admin Admin이(가) 11일 전에 추가함. 약 10시간 전에 수정됨.

상태:	해결	시작시간:	2021/05/28
우선순위:	낮음	완료기한:	
담당자:	-		
Testing:	Code Smells		

설명

위치: GUI_Frame.java / Line: 45

코드의 가독성을 높이기 위해서는 사용하지 않는 함수를 주석처리 대신 삭제해주는 것이 좋다.



6. 주석 처리된 코드

58 결함 해결 **낮음** [Major]: This block of commented-out lines of code should be removed. - Network.java 2021/06/08 00:45 ...

<input type="checkbox"/>	65	결함	해결	낮음	[Major]: This block of commented-out lines of code should be removed. - VMList.java	2021/06/08 00:49	...
<input type="checkbox"/>	64	결함	해결	낮음	[Major]: This block of commented-out lines of code should be removed. - VM.java	2021/06/08 00:48	...
<input type="checkbox"/>	63	결함	해결	낮음	[Major]: This block of commented-out lines of code should be removed. - Card.java	2021/06/08 00:47	...
<input type="checkbox"/>	62	결함	해결	낮음	[Major]: This block of commented-out lines of code should be removed. - GUI_VMList.java	2021/06/08 00:47	...
<input type="checkbox"/>	61	결함	해결	낮음	[Major]: This block of commented-out lines of code should be removed. - GUI_Frame.java	2021/06/08 00:47	...
<input type="checkbox"/>	60	결함	해결	낮음	[Major]: This block of commented-out lines of code should be removed. - Controller.java	2021/06/08 00:46	...

[Major]: This block of commented-out lines of code should be removed. - GUI_VMList.java

admin Admin이(가) 11일 전에 추가함. 약 10시간 전에 수정됨.

상태:	해결	시작시간:	2021/05/28
우선순위:	낮음	완료기한:	
담당자:	-		
Testing:	Code Smells		

설명

위치: GUI_VMList.java / Line: 33

코드의 가독성을 높이기 위해서는 사용하지 않는 함수를 주석처리 대신 삭제해주는 것이 좋다.



6. 주석 처리된 코드

<input type="checkbox"/>	58	결함	해결	낮음	[Major]: This block of commented-out lines of code should be removed. - Network.java	2021/06/08 00:45	...
<input type="checkbox"/>	65	결함	해결	낮음	[Major]: This block of commented-out lines of code should be removed. - VMList.java	2021/06/08 00:49	...
<input type="checkbox"/>	64	결함	해결	낮음	[Major]: This block of commented-out lines of code should be removed. - VM.java	2021/06/08 00:48	...
<input type="checkbox"/>	63	결함	해결	낮음	[Major]: This block of commented-out lines of code should be removed. - Card.java	2021/06/08 00:47	...
<input type="checkbox"/>	62	결함	해결	낮음	[Major]: This block of commented-out lines of code should be removed. - GUI_VMList.java	2021/06/08 00:47	...
<input type="checkbox"/>	61	결함	해결	낮음	[Major]: This block of commented-out lines of code should be removed. - GUI_Frame.java	2021/06/08 00:47	...
<input type="checkbox"/>	60	결함	해결	낮음	[Major]: This block of commented-out lines of code should be removed. - Controller.java	2021/06/08 00:46	...

[Major]: This block of commented-out lines of code should be removed. - Card.java

admin Admin이(가) 11일 전에 추가함. 약 10시간 전에 수정됨.

상태:	해결	시작시간:	2021/05/28
우선순위:	낮음	완료기한:	
담당자:	-		
Testing:	Code Smells		

설명

위치: Card.java / Line:27-40

코드의 가독성을 높이기 위해서는 사용하지 않는 함수를 주석처리 대신 삭제해주는 것이 좋다.



6. 주석 처리된 코드

58 결함 해결 낮음 [Major]: This block of commented-out lines of code should be removed. - Network.java 2021/06/08 00:45 ...

<input type="checkbox"/>	65	결함	해결	낮음	[Major]: This block of commented-out lines of code should be removed. - VMList.java	2021/06/08 00:49	...
<input type="checkbox"/>	64	결함	해결	낮음	[Major]: This block of commented-out lines of code should be removed. - VM.java	2021/06/08 00:48	...
<input type="checkbox"/>	63	결함	해결	낮음	[Major]: This block of commented-out lines of code should be removed. - Card.java	2021/06/08 00:47	...
<input type="checkbox"/>	62	결함	해결	낮음	[Major]: This block of commented-out lines of code should be removed. - GUI_VMList.java	2021/06/08 00:47	...
<input type="checkbox"/>	61	결함	해결	낮음	[Major]: This block of commented-out lines of code should be removed. - GUI_Frame.java	2021/06/08 00:47	...
<input type="checkbox"/>	60	결함	해결	낮음	[Major]: This block of commented-out lines of code should be removed. - Controller.java	2021/06/08 00:46	...

[Major]: This block of commented-out lines of code should be removed. - VM.java

admin Admin이(가) 11일 전에 추가함. 약 10시간 전에 수정됨.

상태:	해결	시작시간:	2021/05/28
우선순위:	낮음	완료기한:	
담당자:	-		
Testing:	Code Smells		

설명

위치 : VM.java / Line : 11



6. 주석 처리된 코드

58 결함 해결 낮음 [Major]: This block of commented-out lines of code should be removed. - Network.java 2021/06/08 00:45 ...

<input type="checkbox"/>	65	결함	해결	낮음	[Major]: This block of commented-out lines of code should be removed. - VMList.java	2021/06/08 00:49	...
<input type="checkbox"/>	64	결함	해결	낮음	[Major]: This block of commented-out lines of code should be removed. - VM.java	2021/06/08 00:48	...
<input type="checkbox"/>	63	결함	해결	낮음	[Major]: This block of commented-out lines of code should be removed. - Card.java	2021/06/08 00:47	...
<input type="checkbox"/>	62	결함	해결	낮음	[Major]: This block of commented-out lines of code should be removed. - GUI_VMList.java	2021/06/08 00:47	...
<input type="checkbox"/>	61	결함	해결	낮음	[Major]: This block of commented-out lines of code should be removed. - GUI_Frame.java	2021/06/08 00:47	...
<input type="checkbox"/>	60	결함	해결	낮음	[Major]: This block of commented-out lines of code should be removed. - Controller.java	2021/06/08 00:46	...

[Major]: This block of commented-out lines of code should be removed. - VMList.java

admin Admin이(가) 11일 전에 추가함. 약 10시간 전에 수정됨.

상태:	해결	시작시간:	2021/05/28
우선순위:	낮음	완료기한:	
담당자:	-		
Testing:	Code Smells		

설명
 위치 : VMList.java / Line : 33 , 42 , 51-54 , 58 , 80
 코드의 가독성을 높이기 위해서는 사용하지 않는 함수를 주석처리 대신 삭제해주는 것이 좋다.



7. 정적 변수 관련 문제

<input type="checkbox"/>	67	결함	해결	낮음	[Minor] Make this final field static too - GUI_Dispenser.java	2021/06/08 00:50	...
<input type="checkbox"/>	66	결함	해결	낮음	[Minor] Make this final field static too - GUI_Error.java	2021/06/08 00:50	...

[Minor] Make this final field static too - GUI_Error.java

admin Admin이(가) 11일 전에 추가함. 약 10시간 전에 수정됨.

상태:	해결	시작시간:	2021/05/28
우선순위:	낮음	완료기한:	
담당자:	-		
Testing:	Code Smells		

설명

위치: GUI_Error.java / Line:13

선언시 초기화된 상수는 final보다 static final로 선언해야한다.



7. 정적 변수 관련 문제

<input type="checkbox"/>	67	결함	해결	낮음	[Minor] Make this final field static too - GUI_Dispenser.java	2021/06/08 00:50	...
<input type="checkbox"/>	66	결함	해결	낮음	[Minor] Make this final field static too - GUI_Error.java	2021/06/08 00:50	...

[Minor] Make this final field static too - GUI_Dispenser.java

admin Admin이(가) 11일 전에 추가함. 약 10시간 전에 수정됨.

상태:	해결	시작시간:	2021/05/28
우선순위:	낮음	완료기한:	
담당자:	-		
Testing:	Code Smells		

설명

위치 : GUI_Dispenser.java / Line : 14

선언시 초기화된 상수는 final보다 static final로 선언해야한다.

8. 코드 중복

69 결함 해결 보통 [Critical] Define a constant instead of duplicating this literal "맑은 고딕", "Receive Msg:", "Send Msg:", "localhost" 2021/06/08 01:01 ...

[Critical] Define a constant instead of duplicating this literal "맑은 고딕", "Receive Msg:", "Send Msg:", "localhost"

admin Admin이(가) 11일 전에 추가함. 약 10시간 전에 수정됨.

상태:	해결	시작시간:	2021/05/28
우선순위:	보통	완료기한:	
담당자:	-		
Testing:	Code Smells		

설명

위치:
GUI_Admin.java - line40, GUI_CardPayment.java - line46, ...
Network.java - line40
Network.java - line96
Network.java - line112

String literals should not be duplicated

Duplicated string literals make the process of refactoring error-prone, since you must be sure to update all occurrences.

On the other hand, constants can be referenced from many places, but only need to be updated in a single place.

Noncompliant Code Example
 With the default threshold of 3:

```
public void run() {
    prepare("action1"); // Noncompliant - "action1" is duplicated 3 times
    execute("action1");
    release("action1");
}
```

```
@SuppressWarnings("all") // Compliant - annotations are excluded
private void method1() { /* ... */ }
@SuppressWarning("all")
private void method2() { /* ... */ }
```

```
public String method3(String a) {
    System.out.println(" " + a + " "); // Compliant - literal "" has less than 5 characters and is excluded
    return ""; // Compliant - literal "" has less than 5 characters and is excluded
}
```

Compliant Solution
 private static final String ACTION_1 = "action1"; // Compliant

```
public void run() {
    prepare(ACTION_1); // Compliant
    execute(ACTION_1);
    release(ACTION_1);
}
```

Exceptions
 To prevent generating some false-positives, literals having less than 5 characters are excluded.



9. 기타 문제

- 68 결함 해결 낮음 [Minor] Use the opposite operator ("!=") instead. - GUI_VMList.java 2021/06/08 00:51 ...
- 73 결함 해결 보통 [Critical] Add a default case to this switch. 2021/06/08 00:55 ...

[Minor] Use the opposite operator ("!=") instead. - GUI_VMList.java

admin Admin이(가) 11일 전에 추가함. 약 10시간 전에 수정됨.

상태:	해결	시작시간:	2021/05/28
우선순위:	낮음	완료기한:	
담당자:	-		
Testing:	Code Smells		

설명

위치 : GUI_VMList.java / Line : 59

!(address[i]==null) 처럼 복잡한 표현 보다는 address[i] !=null 표현으로 더 단순하게 표현하는 것이 좋다.



9. 기타 문제

<input type="checkbox"/>	68	결함	해결	낮음	[Minor] Use the opposite operator ("!=") instead. - GUI_VMList.java	2021/06/08 00:51	...
<input type="checkbox"/>	73	결함	해결	보통	[Critical] Add a default case to this switch.	2021/06/08 00:55	...

[Critical] Add a default case to this switch.

admin Admin이(가) 11일 전에 추가함. 약 10시간 전에 수정됨.

상태: 해결 시작시간: 2021/05/28
 우선순위: 보통 완료기한:
 담당자: -
 Testing: Code Smells
 설명

위치:
GUI_Error.java - line65
GUI_Frame.java - line139, line174
Network.java - line49

"switch" statements should have "default" clauses

The requirement for a final default clause is defensive programming. The clause should either take appropriate action, or contain a suitable comment as to why no action is taken.

Noncompliant Code Example

```
switch (param) { //missing default clause
case 0:
doSomething();
break;
case 1:
doSomethingElse();
break;
}
```

```
switch (param) {
default: // default clause should be the last one
error();
break;
case 0:
doSomething();
break;
case 1:
doSomethingElse();
break;
}
```

< 뒤로 | 2/35 | 다음

Compliant Solution

```
switch (param) {
case 0:
doSomething();
break;
case 1:
doSomethingElse();
break;
default:
error();
break;
}
```

Exceptions

If the switch parameter is an Enum and if all the constants of this enum are used in the case statements, then no default clause is expected.

Example:

```
public enum Day {
SUNDAY, MONDAY
}
...
switch(day) {
case SUNDAY:
doSomething();
break;
case MONDAY:
doSomethingElse();
break;
}
```



10. GUI 함수들 관련 문제

<input type="checkbox"/>	74	결함	의견	보통	[Critical] Use static access with "javax.swing.SwingConstants" for "CENTER".	2021/06/08 00:56	...
<input type="checkbox"/>	72	결함	의견	보통	[Critical] Add a nested comment explaining why this method is empty, throw an UnsupportedOperationException or complete the implementation.	2021/06/08 00:54	...
<input type="checkbox"/>	70	결함	의견	보통	[Critical] Refactor this method to reduce its Cognitive Complexity	2021/06/08 00:54	...

[Critical] Refactor this method to reduce its Cognitive Complexity

« 뒤로 | 4/35 | 다음 »

admin Admin이(가) 11일 전에 추가함. 약 10시간 전에 수정됨.

상태:	의견	시작시간:	2021/05/28
우선순위:	보통	완료기한:	
담당자:	-		
Testing:	Code Smells		

설명

댓글달기

위치:

GUI_CardPayment.java - line38
 GUI_Prepay.java - line36
 GUI_VerificationCodeMode.java - line28

Cognitive Complexity of methods should not be too high

Cognitive Complexity is a measure of how hard the control flow of a method is to understand. Methods with high Cognitive Complexity will be difficult to maintain.

See

<https://www.sonarsource.com/docs/CognitiveComplexity.pdf>



10. GUI 함수들 관련 문제

<input type="checkbox"/>	74	결함	의견	보통	[Critical] Use static access with "javax.swing.SwingConstants" for "CENTER".	2021/06/08 00:56	...
<input type="checkbox"/>	72	결함	의견	보통	[Critical] Add a nested comment explaining why this method is empty, throw an UnsupportedOperationException or complete the implementation.	2021/06/08 00:54	...
<input type="checkbox"/>	70	결함	의견	보통	[Critical] Refactor this method to reduce its Cognitive Complexity	2021/06/08 00:54	...

[Critical] Add a nested comment explaining why this method is empty, throw an UnsupportedOperationException or complete the implementation.

admin Admin이(가) 11일 전에 추가함. 약 10시간 전에 수정됨.

상태:	의견	시작시간:	2021/05/28
우선순위:	보통	완료기한:	
담당자:	-		
Testing:	Code Smells		

< 뒤로 설명

위치:
 GUI_CardPayment.java - line62, line90, line118, ...
 GUI_Prepay.java - line60, line80, ...
 GUI_VerificationCodeMode.java - line54, line83, ...

Methods should not be empty

There are several reasons for a method not to have a method body:

It is an unintentional omission, and should be fixed to prevent an unexpected behavior in production.
 It is not yet, or never will be, supported. In this case an UnsupportedOperationException should be thrown.
 The method is an intentionally-blank override. In this case a nested comment should explain the reason for the blank override.

Noncompliant Code Example

```
public void doSomething() {
}
```

```
public void doSomethingElse() {
}
```

Compliant Solution
 @Override

```
public void doSomething() {
  // Do nothing because of X and Y.
}
```

@Override

```
public void doSomethingElse() {
  throw new UnsupportedOperationException();
}
```

Exceptions
Default (no-argument) constructors are ignored when there are other constructors in the class, as are empty methods in abstract classes.

```
public abstract class Animal {
  void speak() { // default implementation ignored
  }
}
```




02

수정 진행과정

1. `math.random()` 보안 취약

이력

[developer Developer](#)이(가) [일본](#) 이하 전에 변경

- 상태를(를) [신규](#)에서 [해결](#)(으)로 변경되었습니다.

Security Random 사용으로 수정 완료

`math.random()` 함수 대신
`java.security.SecureRandom()` 함수 사용.

```
Random random = new java.security.SecureRandom();
randCode[i] = (char)((random.nextInt( bound: 10)) + 48);
```

2. printStackTrace() 보안 취약

이력

developer Developer이(가) 약 10시간 전에 변경

- 상태를(를) 신규에서 해결(으)로 변경되었습니다.

printStackTrace()를
System.out.println(e.toString())로 변경

printStackTrace()를
System.out.println(e.toString())로 변경

```

/* vmID setting part */
for(int i=0;i<20;i++){
    newLine = bufferedReader.readLine();
    stock[i] = Integer.parseInt(newLine);
    if(stock[i]>0){
        count++;
    }
}
if(count > 7){
    System.out.println("음료는 7종류만 판매할 수 있습니다.");
    System.exit(status: -1);
}

bufferedReader.close();
fileReader.close();
}catch(Exception e){
    System.out.println(e.toString());
}
}

```

3. Null pointer 참조

이력

admin Admin이(가) 11일 전에 변경

- 제목을(를) *Bug-1 : Need NullPointerException*에서
니다.

developer Developer이(가) 약 10시간 전에 변경

- 상태를(를) *신규*에서 *해결*(으)로 변경되었습니다.

NullPointerException 처리 완료

기존 null 포인터 예외처리가 안된 코드에서
예외처리 수정.

```
if(type!=3 && sendMsg != null) {
    System.out.println(sendPrinter + sendMsg.getSrc_id() + " " + sendMsg.getDst_id() + " " + sendMsg.getType() + " " + sendMsg.getDescription());
}
socket.close();
```

4. Resource closed 문제 (파일 입출력)

이력

admin Admin이(가) 11일 전에 변경

- 제목을(를) *Bug-1 : Resources should be closed*에서 *Bug-2 : Resources*

developer Developer이(가) 약 10시간 전에 변경

- 상태를(를) *신규*에서 *해결*(으)로 변경되었습니다.

다음과 같은 구조로 closeable하게 변경.

```
FileReader fileReader = new FileReader("./TextFiles/CardInfo.txt");
BufferedReader bufferedReader = new BufferedReader(fileReader);
```

```
bufferedReader.close();
fileReader.close();
```

기존 FileReader 객체를 Close 할 수 없게
바로 BufferedReader에 넣었던 코드 수정

```
FileReader fileReader = new FileReader( fileName: "./TextFiles/VMInfo.txt");
BufferedReader bufferedReader = new BufferedReader(fileReader);

bufferedReader.close();
fileReader.close();
```

5. Resource closed 문제 (소켓 관련)

이력

developer Developer이(가) 일분 이하 전에 변경

- 상태를(를) 신규에서 해결(으)로 변경되었습니다.

objectInputStream 객체와
objectOutputStream 객체 생성 후 close() 하지 않는 문제를 해결하였습니다.

```
objectInputStream.close();  
objectOutputStream.close();  
socket.close();
```

objectInputStream과 objectOutputStream을
close() 하지 않아 발생하는 문제 해결.

6. 주식 처리된 코드

이력

developer Developer이(가) 약 10시간 전에 변경

- 상태를(를) 신규에서 해결(으)로 변경되었습니다.

삭제 완료

주식 처리된 코드들 삭제

```

1  /*   public boolean[] receiveStock(int vmID, boolean available){
2      boolean[] availableList = new boolean[10];
3      return availableList;
4  }*/
5
6  /* 어떤 함수인지 불명확함.
7     addOperatingList와 이름을 명확히 구분할 필요가 있음 */
8  /*   public void addVM(int vmID, String address){

```

7. 정적 변수 관련 문제

이력

developer Developer이(가) 약 10시간 전에 변경

- 상태를(를) 신규에서 해결(으)로 변경되었습니다.

수정 완료

```
private static final int ONE_SECOND = 1000;
```

선언시 초기화 된 상수 :
final 대신 static final 사용.

8. 코드 중복

이력

admin Admin이(가) 11일 전에 변경

- 제목을(를) *Define a constant instead of duplicating this literal "앞서 [Critical] Define a constant instead of duplicating this literal "!* (으)로 변경되었습니다.

developer Developer이(가) 약 10시간 전에 변경

- 상태를(를) *신규에서 해결(으)로* 변경되었습니다.

수정 완료

```
String receivePrinter = "Receive Msg";
String sendPrinter = "Send Msg";
String localhost = "localhost";
socket = new Socket(localhost, port: SERVER_PORT_OFFSET + i);
System.out.println(receivePrinter+receivedMsg.getSrc_id()+" "+receivedMsg.getDst_id())
```

“localhost”, “Receive Msg”, “Send Msg” 문자열이 반복되어 하드코딩 되어 있는 부분을 수정.

9. 기타 문제

이력

developer Developer이(가) 약 10시간 전에 변경

수정 완료

developer Developer이(가) 약 10시간 전에 변경

- 상태를(를) 신규에서 해결(으)로 변경되었습니다.

```
if(address[i] != null){
```

수식을 간단하게 변경,
default case 추가.

이력

admin Admin이(가) 11일 전에 변경

- 다음 일감으로부터 복사됨:에 [결함 #72: \[Critical\] Add a nested or complete the implementation of UnsupportedOperationException](#) 추가되었습니다.

admin Admin이(가) 11일 전에 변경

- 다음 일감으로 복사됨:에 [결함 #74: \[Critical\] Use static access modifiers to the fields](#) 추가되었습니다.

developer Developer이(가) 약 10시간 전에 변경

- 상태를(를) 신규에서 해결(으)로 변경되었습니다.

network의
switch문 default 추가 완료.

```
default:
    break;
```




03

CTIP 환경에 대한 소감

박혜린

- 젠킨스나 레드마인 등의 도구를 처음 써보았다. 처음에 낯설고 어려웠지만 이를 통해 팀 간 소통이나 이슈관리 면에서 프로젝트 편의성이 높아짐을 느꼈고, 체계적인 프로그래밍을 경험했다.

변장훈

- 기존에는 기능을 구현하는데만 초점을 두었고 기능들이 잘 돌아가기만 하면 개발이 끝났다고 생각했다. 하지만 이번 CTIP 환경에서의 개발을 통해서 기능의 완성이 끝이 아님을 알게 되었고 코드의 질을 높이려면 어떤 도구를 가지고 어떻게 진행해야 하는지 알게 되었다.

정인욱

- 이슈들을 레드마인이라는 이슈 트래킹 도구에 몰아서 프로젝트를 진행했다. 이로 인해 진행 과정을 한 눈에 볼 수 있었고, 해야 할 일이 얼마나 진행되었는지 파악하기 쉬웠다. CTIP 환경을 이용해 코드를 Github에 올리면 자동으로 코드를 정적 분석해주었다. 이러한 도구를 잘 이용하면 코드의 퀄리티를 쉽게 높일 수 있다.

04 UP 개발 방식에 대한 소감

박혜린

- 규모 있는 프로젝트에서 단계별 설계와 짜임새 있는 문서작업이 중요하다고 느꼈다. requirements부터 시작해서 메서드를 만들기까지 수정과 보완을 거듭하며 참 괴롭기도 했으나 이를 통해 논리적인 개발 과정을 배울 수 있었다.

변장훈

-기존에는 개발을 할때 머릿속으로 생각해서 바로 타이핑을 시작했는데 UP 개발 방식을 사용해 보면서 정해진 프로세스대로 개발을 진행하는 것도 나쁘지 않다고 느꼈다. 특히 문서작업이 매우 귀찮고 힘들었는데 막상 몇주 지나니깐 우리가 초반에 어떻게 생각하고 기능을 명세했는지는 종종 헛갈리는 상황에 놓이게 되었었고 그것을 문서를 통해 복기할수 있어서 기능 구현에 있어서 개발의도나 의견 취합에 있어서 좋았던 것 같다.

정인욱

- 주먹구구식으로 코딩부터 시작하지 않고 OOA부터 OOI까지 정해진 프로세스대로 체계적으로 수행하여 비교적 퀄리티 있는 구현물이 나온 것 같다. OOA, OOD를 얼마나 잘 수행했는지에 따라 구현에 걸리는 시간이 실제로 크게 달라지는 것을 깨닫게 되었다. 여러명이 함께 진행하는 규모 있는 프로젝트에서는 이러한 체계적인 프로세스가 꼭 필요하다는 것을 깨달았다.

감사합니다