

# Software V&V

2<sup>nd</sup> Testing

<Static Analysis & System Test>

1팀

컴퓨터공학부 201711381 김소현

컴퓨터공학부 201711394 민하은

컴퓨터공학부 201711401 염혜지

컴퓨터공학부 201711407 이가현

컴퓨터공학부 201711435 홍예주

# 목차

---

1 2<sup>nd</sup> Specification Review + Stub

2 2<sup>nd</sup> Category Partition Test

3 2<sup>nd</sup> Brute Force Test

4 Static Analysis & Quality

5 Overall Review

**PART 1**

**2<sup>nd</sup> Specification Review**  
**+ Stub**

# Specification Review

## 1. 서비스를 시작한다.

Use Case	1. 서비스를 시작한다.
Actor	System
Type	Hidden
Cross Reference	Functions: R 1.1, R 1.2 Use cases: "서비스를 시작한다", "DVM 선택을 진행한다"
Pre-Requisites	N/A
Typical Courses of Events	(A): Actor, (S): System 1. (S) 화면에 전체 DVM 목록을 보여준다.
Alternative Courses of Events	N/A
Exceptional Courses of Events	N/A

## 2. DVM 선택을 진행한다.

Use Case	2. DVM 선택을 진행한다.
Actor	Customer
Type	Evident
Cross Reference	Functions: R 1.2, R 1.1 Use cases : "DVM 선택을 진행한다.", "서비스를 시작한다."
Pre-Requisites	N/A
Typical Courses of Events	(A): Actor, (S): System 1.(A) Customer 가 사용하고자 하는 DVM 을 선택한다. 2.(S) 선택된 DVM 시스템을 사용한다. 3.(S) 음료 메뉴를 보여준다.
Alternative Courses of Events	N/A
Exceptional Courses of Events	N/A

- ➔
- 1번 "서비스를 시작한다"에 use case "DVM을 선택한다"가 들어갈 필요가 없다. 왜냐하면 2번이 DVM 선택을 진행한다 이기 때문이다.

# Specification Review

## 3. 현재 자판기에 판매하는 음료수를 선택한다

<b>Use Case</b>	3. 현재 자판기에서 판매하는 음료수를 선택한다.
<b>Actor</b>	Customer
<b>Type</b>	Evident
<b>Cross Reference</b>	Functions: R1.2, R2.1, R3.1, R4.1 Use cases : "DVM 선택을 진행한다", "현재 자판기에 판매하는 음료수를 선택한다", "결제 방법 중 하나를 선택한다", "현재 자판기의 재고를 확인한다"
<b>Pre-Requisites</b>	- 전체 20가지의 음료 종류를 알아야 한다.
<b>Typical Courses of Events</b>	(A1): Actor, (S): System 1. (A1) Customer 가 현재 자판기에 판매하는 음료수를 선택한다. 2. (S) "현재 자판기의 재고를 확인한다"를 invoke하여 선택한 음료수의 재고를 확인한다. 3. (S) "결제 방법 중 하나를 선택한다"를 invoke 한다.
<b>Alternative Courses of Events</b>	N/A
<b>Exceptional Courses of Events</b>	Line2. 현재 자판기의 재고가 없다면 재고가 없다는 메시지를 출력 후 "선결제를 진행한다"를 invoke한다.



- Ref 1.2 DVM을 선택한다 지워야 한다. → usecase 2번에 해당한다.
- Ref 3.1 결제 방법 중 하나를 선택한다 지워야 한다.  
→ usecase 5번에 해당한다.

# Specification Review

## 4. 현재 자판기에서 판매하지 않는 음료수를 선택한다

<b>Use Case</b>	4. 현재 자판기에서 판매하지 않는 음료수를 선택한다.
<b>Actor</b>	Customer
<b>Type</b>	Evident
<b>Cross Reference</b>	Functions: R1.2, R2.2, R6.1 Use cases : "DVM 선택을 진행한다","현재 자판기에서 판매하지 않는 음료수를 선택한다","선결제를 진행한다"
<b>Pre-Requisites</b>	- 전체 20가지의 음료 종류를 알아야 한다.
<b>Typical Courses of Events</b>	(A1): Actor, (S): System 1. (A1) Customer 가 현재 자판기에서 판매하지 않는 음료수를 선택한다.. 2. (S) "선결제를 진행한다"를 invoke한다.
<b>Alternative Courses of Events</b>	N/A
<b>Exceptional Courses of Events</b>	N/A



- Ref 1.2 지워야 한다.
  - Ref 6.1 지워야 한다.
- usecase 12번에 해당한다.

# Specification Review

## 6. 카드 결제를 진행한다

<b>Use Case</b>	6. 카드 결제를 진행한다.
<b>Actor</b>	Customer
<b>Type</b>	Evident
<b>Cross Reference</b>	Functions: R3.2, R3.4, R6.1 Use cases : "카드 결제를 진행한다", "결제를 취소한다", "선결제를 진행한다"
<b>Pre-Requisites</b>	- Customer 가 결제 방법 중 카드 결제를 선택해야 한다.
<b>Typical Courses of Events</b>	(A): Actor, (S): System 1. (A) Customer 는 카드 결제를 위해, 카드를 결제 단말기에 꽂는다. 2. (S) Customer 가 결제 단말기에 사용한 카드가 결제를 할 수 있는 종류의 카드인지 확인한다. 3. (S) 결제할 음료의 가격을 전달받는다. 4. (S) 결제 단말기에 사용한 카드의 잔액이 음료의 가격을 결제할 수 있는지 확인한다. 5. (S) 카드 결제를 완료한다. 6. (S) 재고를 업데이트한다.
<b>Alternative Courses of Events</b>	N/A
<b>Exceptional Courses of Events</b>	Line 2. 카드가 결제를 할 수 없는 종류의 카드인 경우, "결제를 취소한다."를 invoke 한다. Line 3. 카드의 잔액이 음료의 가격을 결제할 수 없을 경우, "결제를 취소한다"를 invoke 한다.



- Ref 6.1 선결제를 진행한다 지워야 한다.

→ usecase 12번에 해당할 뿐더러 event 진행 순서에 선결제에 관한 이야기도 없다.

.

## DVM stub 구축 및 소통

**예주** **홍예주** 오전 11:57  
스텝 개발 진행상황은 어떻게 되어가고 있나요?

**예주** **4개의 답글** 4일 전 마지막 답글

**이가현** 오후 5:19  
+ 레드마인에 올라온 결함 사항들도 수정 진행중이신가요 ?!

**이가현** **4개의 답글** 5일 전 마지막 답글

**예주** **홍예주** 오후 5:50  
스레드에 답글을 남김: **스텝 개발 진행상황은 어떻게 되어가고 있나요?**  
이번주 목->금 넘어가는 자정까지 수정코드와 스텝이 깃에 올라와야 합니다~!  
수고해주세요!  
[최근 댓글 보기](#)



**예주** **홍예주** 오후 2:45  
수정한 스텝 파일(jar) 슬랙에 따로 올려주세요!

**신원세** 오후 2:48  
Zip ▾

 **DVMs.zip**  
18MB Zip



수정한 스텝 파일입니다!





The screenshot displays a Windows desktop with several command prompt windows. The windows show the execution of Java programs (MainController.jar, DVM1.jar, DVM2.jar, DVM3.jar, DVM4.jar) and their logs. The logs indicate a sequence of messages being sent and received between a Controller and various DVMs (Distributed Virtual Machines).

Log for MainController.jar:

```

C:\Users\judy8>cd C:\Users\judy8\Downloads\DVMs
C:\Users\judy8\Downloads\DVMs>java -jar MainController.jar
[Controller] DVM1에게 메시지 발신(유형: 4(위치 요청))
[Controller] DVM1으로부터 메시지 수신(유형: 5(위치 응답), 내용: 101)
[Controller] DVM2에게 메시지 발신(유형: 4(위치 요청))
[Controller] DVM2으로부터 메시지 수신(유형: 5(위치 응답), 내용: 202)
[Controller] DVM3에게 메시지 발신(유형: 4(위치 요청))
[Controller] DVM3으로부터 메시지 수신(유형: 5(위치 응답), 내용: 303)
[Controller] DVM4에게 메시지 발신(유형: 4(위치 요청))
[Controller] DVM4으로부터 메시지 수신(유형: 5(위치 응답), 내용: 404)
[Controller] DVM5에게 메시지 발신(유형: 4(위치 요청))
[Controller] DVM5으로부터 메시지 수신(유형: 5(위치 응답), 내용: 505)
[Controller] DVM6에게 메시지 발신(유형: 4(위치 요청))
[Controller] DVM6으로부터 메시지 수신(유형: 5(위치 응답), 내용: 606)
[Controller] DVM7에게 메시지 발신(유형: 4(위치 요청))
[Controller] DVM7으로부터 메시지 수신(유형: 5(위치 응답), 내용: 707)
[Controller] DVM8에게 메시지 발신(유형: 4(위치 요청))
[Controller] DVM8으로부터 메시지 수신(유형: 5(위치 응답), 내용: 808)

```

Log for DVM1.jar:

```

C:\Users\judy8\Downloads\DVMs>java -jar DVM1.jar
[DVM1] SERVER ON
[DVM1] DVM1로부터 메시지 수신(유형: 4, 내용: 101)
[DVM1] Controller에게 메시지 발신(유형: 5(위치 응답), 내용: 101)

```

Log for DVM2.jar:

```

C:\Users\judy8\Downloads\DVMs>java -jar DVM2.jar
[DVM2] SERVER ON
[DVM2] DVM2로부터 메시지 수신(유형: 4, 내용: 202)
[DVM2] Controller에게 메시지 발신(유형: 5(위치 응답), 내용: 202)

```

Log for DVM3.jar:

```

C:\Users\judy8\Downloads\DVMs>java -jar DVM3.jar
[DVM3] SERVER ON
[DVM3] DVM3로부터 메시지 수신(유형: 4, 내용: 303)
[DVM3] Controller에게 메시지 발신(유형: 5(위치 응답), 내용: 303)

```

Log for DVM4.jar:

```

C:\Users\judy8\Downloads\DVMs>java -jar DVM4.jar
[DVM4] SERVER ON
[DVM4] DVM4로부터 메시지 수신(유형: 4, 내용: 404)
[DVM4] Controller에게 메시지 발신(유형: 5(위치 응답), 내용: 404)

```

On the right side of the screenshot, there is a window showing a grid of buttons. The buttons are arranged in two rows of four columns. The top row contains buttons labeled '1. DVM1 주소: 101', '2. DVM2 주소: 202', '3. DVM3 주소: 303', and '4. DVM4 주소: 404'. The bottom row contains buttons labeled '5. DVM5 주소: 505', '6. DVM6 주소: 606', '7. DVM7 주소: 707', and '8. DVM8 주소: 808'. Below the grid, there is a button labeled '확인' (Confirm).

- 1<sup>st</sup> Cycle에서 구현하지 못한 Stub



- 소켓통신을 이용하여 구현한 Stub 받음.
- 직접 실행해본 결과 제대로 통신이 되고 있는 것 확인.

**PART 2**

**2<sup>nd</sup> Category Partition Test**

# 2<sup>nd</sup> Category Partition Test

Group	Category	Value	constraint	key	
DVM_number	select DVM number	1>= and <=8		1	
		<1	error	2	
		>8	error	3	
		1 to 8 with a leading zero		4	
		not (1 to 8) with a leading zero	error	5	
Drink	select Drink	1~7 with stock	property cs	1	
		8~20 with stock	property cns	2	
		<1 or >20	error	3	
Payment_select	select Payment	1	if cs   property card	1	
		1 next 2	if cs   property code	2	
		first 2	error	3	
		exclude 1,2	error	4	
Payment_process	card payment (선결제가 아닌 카드결제)	valid, 잔고 o	if card   property card_s	1	
		valid, 잔고 x	if card	2	
		8자리인데 invalid	error	3	
		8자리가 아닌경우	error	4	
	pre-payment (선결제인데 카드결제)	valid, 잔고 o	if cns   property pre	1	
		valid, 잔고 x	if cns	2	
		8자리인데 invalid	error	3	
		8자리가 아닌경우	error	4	
	get_code		get valid code	if pre	1
			get invalid code	error	2
			get no code	error	3
	code payment (선결제하고나서 코드결제)		valid한 코드	if code   property code_s	1
			5자리의 invalid코드	error	2
5자리가 아닌 경우			error	3	
Environment	number of DVM	8		1	
		not 8	error	2	
	number of Drink types	20		1	
		not 20	error	2	
	DVM drink stock	not zero		1	
both zero		single	2		
Drink_output	drink_output	card valid output	if card_s	1	
		code valid output	if code_s	2	
		invalid output	error	3	

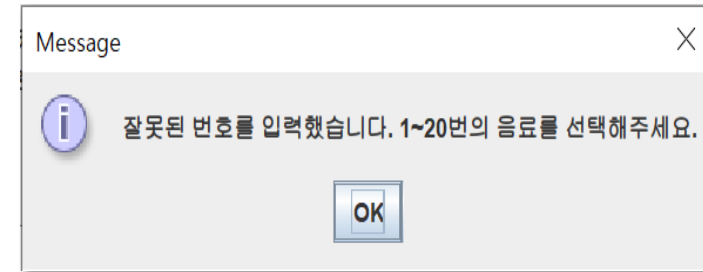
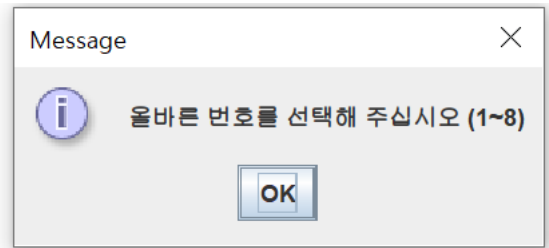
- Category를 구분하여 Test Case를 작성해본 결과,  
 $5 \times 3 \times 4 \times 4 \times 4 \times 3 \times 3 \times 2 \times 2 \times 2 \times 3 = 207360$  개
- 개발과정에서의 차이를 보기 위하여 똑같은 category로 category partition test를 진행하였다.

# Category Partition Test

7

FAIL

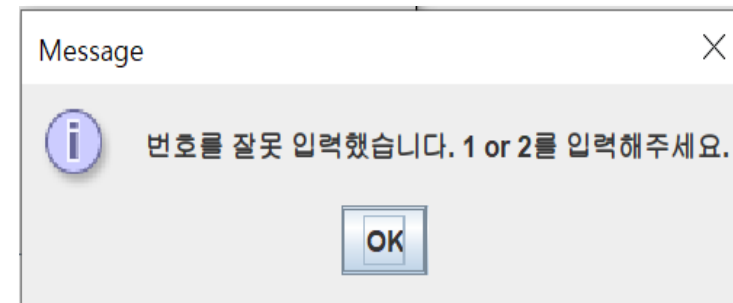
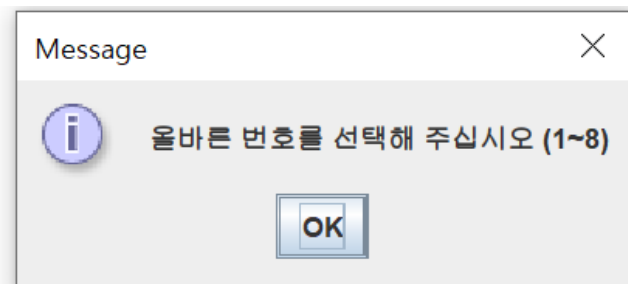
음료선택창에서 0을 입력하면 1~20사이의 번호를 선택해달라고 하지 않고, 1~8사이의 번호를 선택해달라는 안내문이 뜬다



9

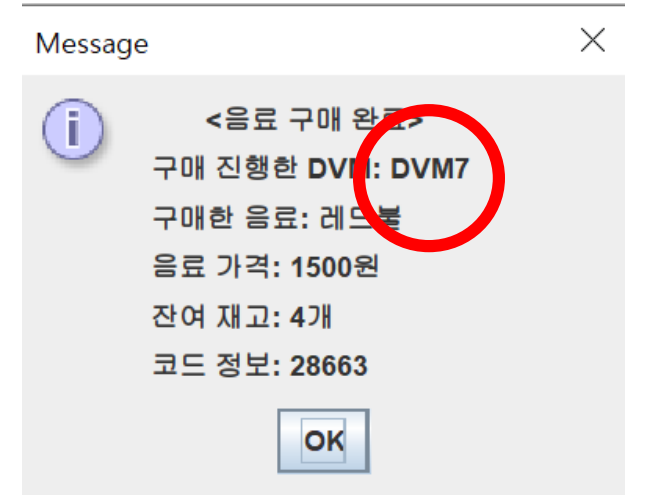
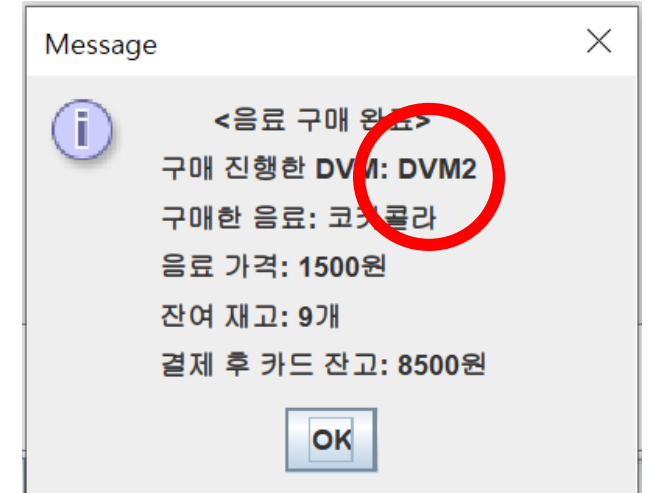
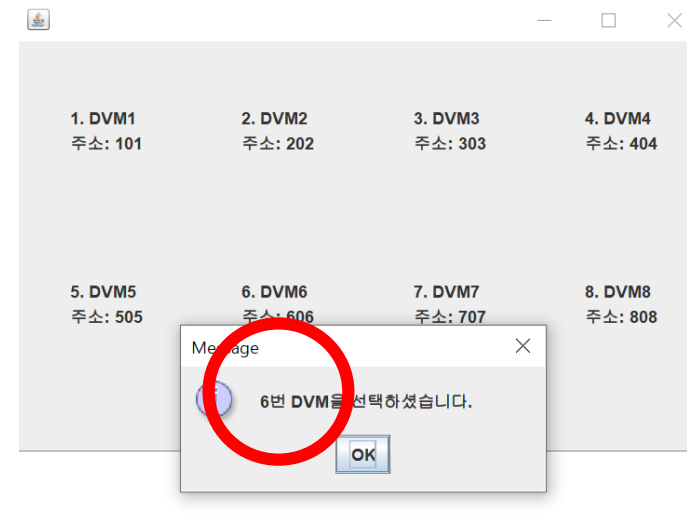
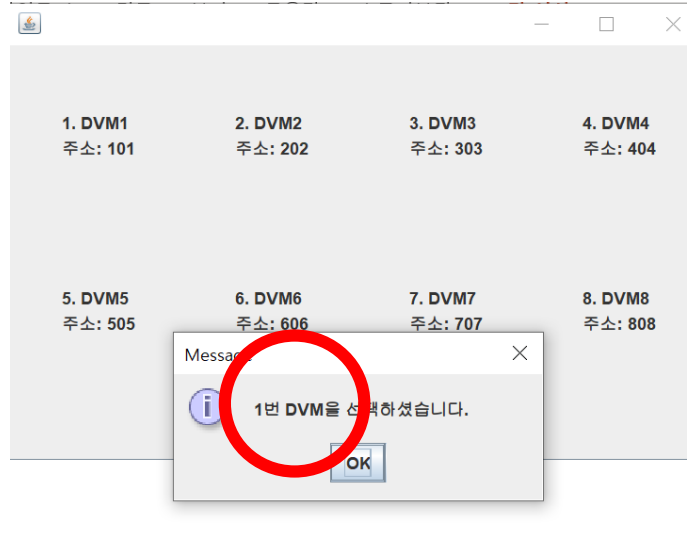
FAIL

결제선택창에서 0을 입력하면 1과 2 중 선택해달라고 하지 않고, 1~8사이의 번호를 선택해달라는 안내문이 뜬다



# Category Partition Test





번호	expected output	test result	비고
1	전체 DVM 수가 8개	pass	
2	전체 음료 종류가 20개	pass	
3	모든 DVM에 재고가 없으면 재고가 없다고 알려줘야 함	pass	
4	DVM의 번호가 1 미만인 것을 고를 수 없음	pass	
5	DVM의 번호가 8 초과인 것을 고를 수 없음	pass	
6	0으로 시작하는 1과 8 사이의 숫자가 아닌 DVM 번호를 고를 수 없음	pass	
7	음료 번호를 1 미만이나 20 초과로 고를 수 없음	pass	
8	선택제를 하지 않고 코드 결제를 할 수 없음	pass	
9	결제 선택에서 1과 2를 제외한 숫자를 고를 수 없음	pass	
10	카드 번호가 8자리라도 유효하지 않으면 결제할 수 없음	pass	
11	카드 결제 시 카드 번호가 8자리가 아니면 결제할 수 없음	pass	
12	선 결제 시 카드번호가 8자리라도 유효하지 않으면 결제할 수 없음	pass	
13	선결제 시 카드번호가 8자리가 아니면 결제할 수 없음	pass	
14	선결제 시 유효하지 않은 코드가 나오면 안됨	pass	
15	선결제 시 코드가 나오지 않으면 안됨	pass	
16	코드 결제 시 코드가 5자리지만 유효한 코드 아니면 결제가 되지 않음	pass	
17	코드 결제 시 코드가 5자리가 아니면 결제가 되지 않음	pass	
18	음료가 유효하지 않게 출력되면 안됨	pass	
19	잔액이 없는 카드로 결제를 시도해서 실패함	pass	
20	잔액이 있는 카드로 올바르게 결제 진행하여 올바른 음료가 나옴	fail	구매한 dvm의 번호가 잘못나옴
21	선결제 후 올바른 코드로 결제하여 올바른 음료가 나옴	fail	구매한 dvm의 번호가 잘못나옴
22	잔액이 없는 카드로 선결제를 시도하여 선결제 실패함	pass	
23	선결제는 성공하여 올바른 코드 5자리를 받음	pass	
24	잔고가 없는 카드를 이용하여 결제 시도 후 음료를 받지 못함	pass	
25	잔고가 있는 카드를 이용하여 카드결제 시도 후 음료를 성공적으로 받음	fail	구매한 dvm의 번호가 잘못나옴
26	선결제 완료 후 받은 코드번호를 사용하여 원하는 음료를 구매한다	fail	구매한 dvm의 번호가 잘못나옴
27	현재 DVM에 없는 음료를 선택하여 선결제를 진행 시 카드 결제를 진행한다	pass	
28	현재 DVM에 없는 음료를 선택하여 선결제 진행 후 올바른 5자리 코드를 받는다	pass	





CPT pass percentage = 85.7%

# Category Partition Test

## TestLink





-  **V&V-57:[2nd cycle] tc20**
-  **V&V-58:[2nd cycle] tc21**
-  **V&V-59:[2nd cycle] tc25**
-  **V&V-60:[2nd cycle] tc26**


V&V-58 : [2nd cycle] tc21 - Version1  

**Preconditions**










1. DVM이 1번부터 8번까지 8종류가 있다.
2. 음료의 종류가 20가지이다.
3. 모든 DVM 중 하나 이상에는 원하는 음료의 재고가 존재한다.
4. 선결제 후 올바른 코드를 받아놓아야 한다.

**Summary**

#	Step actions	Expected Results	Execution	
1	받은 유효한 5자리 코드를 입력한다.	올바른 코드를 입력했다고 뜬다.	Manual	 
2	올바른 음료가 나온다.	올바른 음료를 구매했다고 창이 뜬다.	Manual	 

Date : 05/28/2021 06:24:46 - Tested by : admin - Build : category partitioning test build - Status : Failed 

last execution (current build) - Build : category partitioning test build

Date	Build	Tested by	Status	Exec (min)	Version	BUG management	Run mode	
 05/28/2021 06:24:46	category partitioning test build	admin	<b>Failed</b>		1	 	  	
<b>Build</b>		<b>Test Case Step</b>	<b>Relevant bugs</b>					
category partitioning test build		126 : [신규] : [2nd Cycle] [CPT - Test Case 21] 구매한 DVM 번호가 잘못나옴						  

# Category Partition Test

## Redmine

<input type="checkbox"/>	128	결함	신규	보통	[2nd Cycle] [CPT - Test Case 26] 구매한 DVM 번호가 잘못나옴	2021/05/29 00:26	...
<input type="checkbox"/>	127	결함	신규	보통	[2nd Cycle] [CPT - Test Case 25] 구매한 DVM 번호가 잘못나옴	2021/05/29 00:30	...
<input type="checkbox"/>	126	결함	신규	보통	[2nd Cycle] [CPT - Test Case 21] 구매한 DVM 번호가 잘못나옴	2021/05/29 00:25	...
<input type="checkbox"/>	125	결함	신규	보통	[2nd Cycle] [CPT - Test Case 20] 구매한 DVM 번호가 잘못나옴	2021/05/29 00:25	...

### 결함 #126

 편집  작업시간 기록  지켜보기  복사  삭제


#### [2nd Cycle] [CPT - Test Case 21] 구매한 DVM 번호가 잘못나옴

« 뒤로 | 114/117 | 다음 »

HyeJi Yeom이(가) 약 15시간 전에 추가함. 약 15시간 전에 수정됨.

상태:	신규	시작시간:	2021/05/28
우선순위:	보통	완료기한:	
담당자:	-	진척도:	<div style="width: 0%;"></div> 0%
범주:	[2nd Cycle] Category Partitioning Test	추정시간:	

#### 설명

 댓글달기

1) 선결제 후 올바른 코드로 결제하여 올바른 음료가 나오는지 테스트  
-> 구매한 DVM 번호가 잘못나옴

#### 하위 일감

[추가](#)

#### 연결된 일감

[추가](#)

#### 이력

admin Admin이(가) 약 24시간 전에 변경

   #1

Link to TestLink Execution: [http://54.180.204.228:81/testlink/ltx.php?item=exec&feature\\_id=58&build\\_id=5](http://54.180.204.228:81/testlink/ltx.php?item=exec&feature_id=58&build_id=5)

Link to TestLink Execution Print View: <http://54.180.204.228:81/testlink/lnl.php?type=exec&id=80&apikey=e690960a602675a162c0361c63f6e8df1fa8df50cbba9309f5b57b841d4fc9d4>

**PART 3**

**2<sup>nd</sup> Brute Force Test**



# Brute Force Test

Test	Test Num	Description
Select drink test	1-1	선택한 음료로 결제 진행이 잘 되는지 확인한다.
	1-2	없는 번호의 음료를 선택한다.
	1-3	정해진 가격대로 정확히 출력되는지 확인한다.
Proceed Card-Payment test	2-1	사용 가능한 카드를 입력한다.
	2-2	사용 불가능한 카드를 입력한다.
	2-3	카드의 잔액이 충분할 때만 결제가 진행되는지 확인한다.
	2-4	카드의 잔액이 부족할 경우 잔액 부족 메시지를 출력하는지 확인한다.
Check current machine stock test	3-1	현재 자판기의 재고가 정확한지 확인한다.
	3-2	현재 자판기에서 재고가 0개일 때 재고가 부족하다는 메시지를 출력하는지 확인한다.
Check other DVMs stock test	4-1	자기 외의 다른 자판기의 재고가 정확하게 표시되는지 확인한다.
DVMs location test	5-1	재고가 있는 자판기의 위치를 올바르게 출력하는지 확인한다.
Make code test	6-1	랜덤으로 생성된 인증코드가 기존의 인증코드와 중복되지 않는지 확인한다.
	6-2	화면에 출력되는 인증코드가 사용자가 선결제해서 생성된 인증코드와 일치하는지 확인한다.
Enter code test	7-1	코드를 생성한 후 생성한 코드와 같은 코드를 입력해본다.
	7-2	코드를 생성한 후 생성한 코드와 다른 코드를 입력해본다.
	7-3	생성한 코드가 없는 채로 코드 입력을 시도해본다.

No	P/F	Expected Output	Test result
1-1	P	선택한 음료에 대한 결제가 성공적으로 진행된다.	선택한 음료에 대한 결제가 성공적으로 진행된다.
1-2	P	잘못된 번호입니다 라는 메시지가 출력된다.	잘못된 번호입니다 라는 메시지가 출력된다.
1-3	P	선택한 음료에 대한 가격 정확히 출력된다.	선택한 음료에 대한 가격 정확히 출력된다.
2-1	P	카드결제가 성공적으로 진행된다.	카드결제가 성공적으로 진행된다.
2-2	P	에러 메시지가 뜬 후 초기화면으로 돌아간다.	에러 메시지가 뜬 후 초기화면으로 돌아간다.
2-3	P	카드결제가 성공적으로 진행되고 음료가 가격만큼 잔액이 차감된다.	카드결제가 성공적으로 진행되고 음료가 가격만큼 잔액이 차감된다.
2-4	P	잔액부족 메시지를 출력한다.	잔액부족 메시지를 출력한다.
3-1	P	dvm의 음료가 재고 개수만큼 구매된다.	dvm의 음료가 재고 개수만큼 구매된다.
3-2	P	재고없음 문구가 뜨고 초기화면으로 돌아간다.	재고없음 문구가 뜨고 초기화면으로 돌아간다.
4-1	P	다른 dvm의 음료가 재고 개수만큼 선결제 된다.	다른 dvm의 음료가 재고 개수만큼 선결제 된다.
5-1	P	DVM 번호 옆에 주소가 101 202 형태로 뜬다.	DVM 번호 옆에 주소가 101 202 형태로 뜬다.
6-1	P	메세지로 출력되는 인증코드가 이전과 겹치지 않는다.	메세지로 출력되는 인증코드가 이전과 겹치지 않는다.
6-2	P	선결제 후 출력되는 인증코드 번호와 코드결제 후 출력되는 인증코드 번호가 동일하다.	선결제 후 출력되는 인증코드 번호와 코드결제 후 출력되는 인증코드 번호가 동일하다.
7-1	P	코드결제가 성공적으로 진행된다.	코드결제가 성공적으로 진행된다.
7-2	P	유효하지 않는 인증코드입니다 라는 메시지가 출력된 후 초기화면으로 돌아간다.	유효하지 않는 인증코드입니다 라는 메시지가 출력된 후 초기화면으로 돌아간다.
7-3	P	유효하지 않는 인증코드입니다 라는 메시지가 출력된 후 초기화면으로 돌아간다.	유효하지 않는 인증코드입니다 라는 메시지가 출력된 후 초기화면으로 돌아간다.

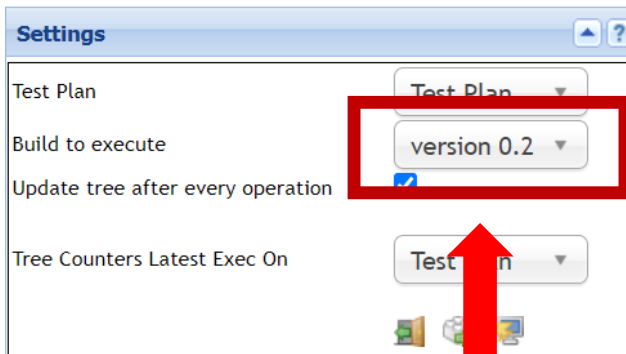
# Brute Force Test

Test	Test Num	Description
Check code test	8-1	유효하지 않은 인증코드를 입력해본다.
	8-2	이미 사용한 인증코드를 입력해본다.
Check drink test	9-1	제공할 음료가 사용자가 결제한 음료와 일치하는지 확인한다.
Provide drink test	10-1	음료가 정확히 사용자에게 제공되는지 확인한다.
Check pay result	11-1	코드 결제 후 음료 구매 완료에 대한 메시지가 올바르게 출력되는지 확인한다.

**BFT pass percentage = 95.24%**

No	P/F	Expected Output	Test result
8-1	P	유효하지 않는 인증코드입니다 라는 메시지가 출력된 후 초기화면으로 돌아간다.	유효하지 않는 인증코드입니다 라는 메시지가 출력된 후 초기화면으로 돌아간다.
8-2	P	유효하지 않는 인증코드입니다 라는 메시지가 출력된 후 초기화면으로 돌아간다.	유효하지 않는 인증코드입니다 라는 메시지가 출력된 후 초기화면으로 돌아간다.
9-1	P	Dvm에서 선택한 음료와 구매 후 출력되는 음료가 같다.	Dvm에서 선택한 음료와 구매 후 출력되는 음료가 같다.
10-1	P	Dvm에서 선택한 음료와 구매 후 출력되는 음료가 같다.	Dvm에서 선택한 음료와 구매 후 출력되는 음료가 같다.
11-1	F	음료를 구매한 dvm 번호와 구매완료 메시지로 출력된 dvm번호가 같아야한다.	Dvm번호가 같지 않다.

## TestLink

2<sup>nd</sup> Cycle

Test Results on Build version 0.2

Print Show complete execution history Import XML Results

Test Plan Test Plan

Build version 0.2

Test Suite : check pay result/

Last execution (any build)

Date : 05/30/2021 11:37:33 - Tested by : admin - Build : version 0.2 - Status : Failed

Last execution (current build) - Build : version 0.2

Date	Build	Tested by	Status	Exec (min)	Version	BUG management	Run mode
05/30/2021 11:37:33	version 0.2	admin	Failed		1		
	Build	Test Case Step	Relevant bugs				
	version 0.2	237 : [신-카] : [2nd Cycle] [BFT] 음료 구매 완료 메시지 출력 확인					

Redmine의 결함 번호와 연계

## Redmine

결함 #237

 편집
  작업시간 기록
  지켜보기
  복사
  삭제


## [2nd Cycle] [BFT] 음료 구매 완료 메시지 출력 확인

[« 뒤로](#) | 1/109 | [다음 >](#)

admin Admin이(가) 4일 전에 추가함. 2일 전에 수정됨.

상태:	신규	시작시간:	2021/05/28
우선순위:	보통	완료기한:	
담당자:	-	진척도:	<div style="width: 0%;"></div> 0%
범주:	brute force testing	추정시간:	

## 설명

 댓글달기

결제 후 음료 구매 완료에 대한 메시지가 올바르게 출력되는지 확인한다.  
=> 구매 완료된 DVM 번호 오류

## 하위 일감

[추가](#)

## 연결된 일감

[추가](#)

## 이력

admin Admin이(가) 2일 전에 변경

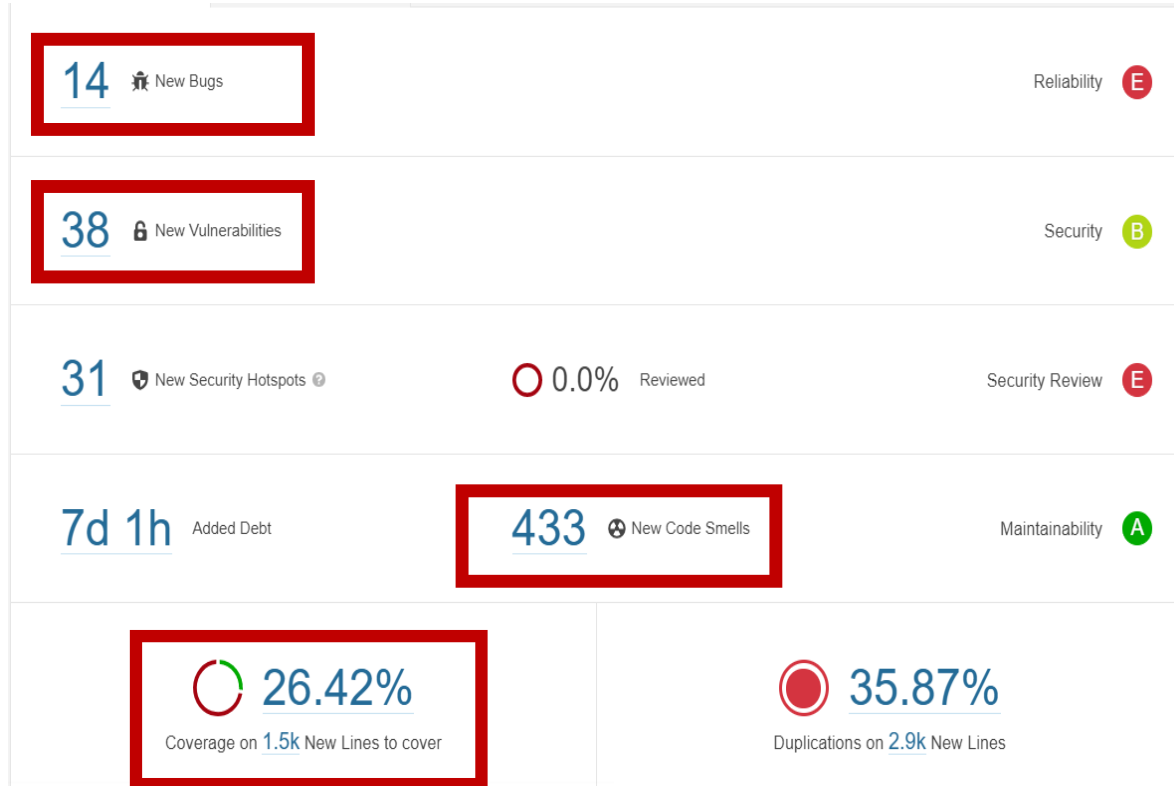


 #1
Link to TestLink Execution: [http://54.180.204.228:81/testlink/ltx.php?item=exec&feature\\_id=62&build\\_id=6](http://54.180.204.228:81/testlink/ltx.php?item=exec&feature_id=62&build_id=6)

# PART 4

# Static Analysis & Quality

# Quality - 1. Sonarway



- Bugs : 14
- Code smell : 433
- Vulnerability : 38
- Coverage : 26.42%

-> 전체 447개의 버그 및 code smell 중 29개의 유의미한 문제점 추려냄

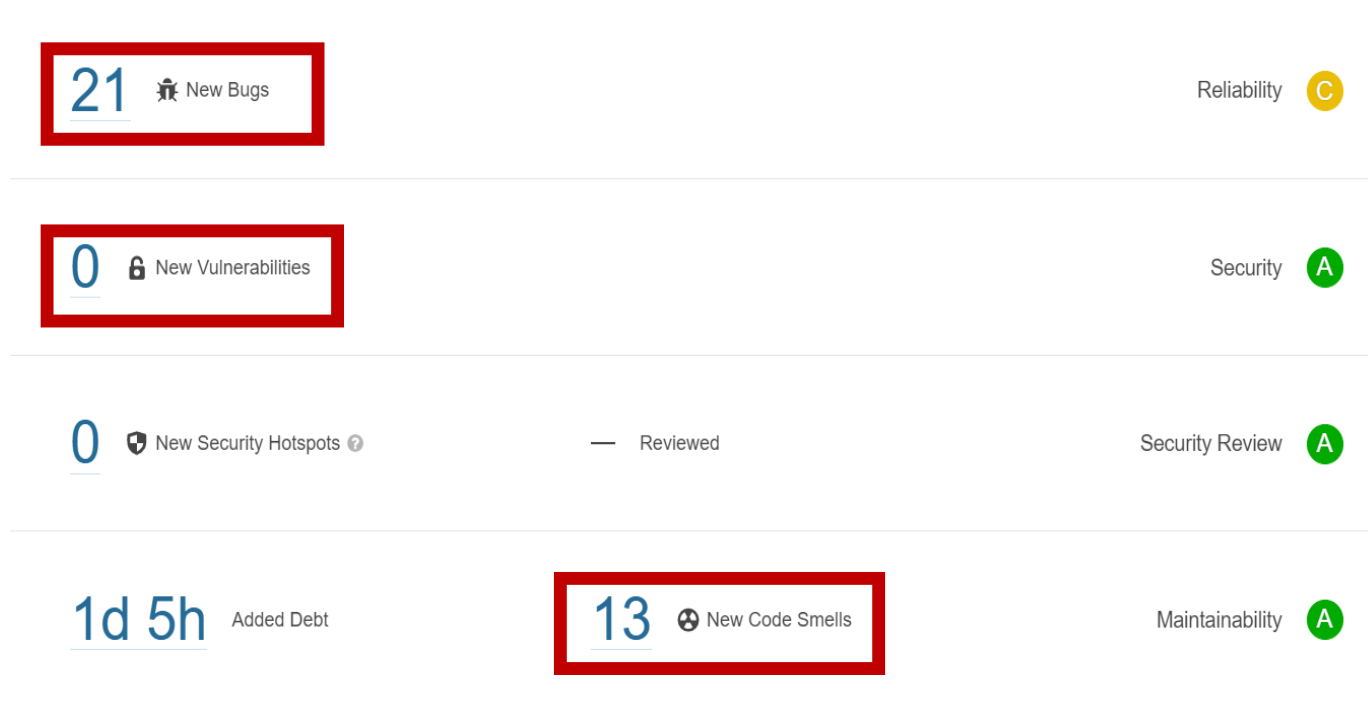
# Quality - 1. Sonarway

File <sup>↵</sup>	Line <sup>↵</sup>	Redmine issue# <sup>↵</sup>	Review <sup>↵</sup>
DVM1 <sup>↵</sup>	75 <sup>↵</sup>	130 <sup>↵</sup>	Add a default case to this switch <sup>↵</sup>
	54 <sup>↵</sup>	152 <sup>↵</sup>	Call the method Thread.start() to execute the content of the run() method in a dedicated thread. <sup>↵</sup>
DVM2 <sup>↵</sup>	71 <sup>↵</sup>	133 <sup>↵</sup>	Add a default case to this switch <sup>↵</sup>
	53 <sup>↵</sup>	155 <sup>↵</sup>	Call the method Thread.start() to execute the content of the run() method in a dedicated thread. <sup>↵</sup>
DVM3 <sup>↵</sup>	71 <sup>↵</sup>	134 <sup>↵</sup>	Add a default case to this switch <sup>↵</sup>
	53 <sup>↵</sup>	161 <sup>↵</sup>	Call the method Thread.start() to execute the content of the run() method in a dedicated thread. <sup>↵</sup>
DVM4 <sup>↵</sup>	71 <sup>↵</sup>	135 <sup>↵</sup>	Add a default case to this switch <sup>↵</sup>
	53 <sup>↵</sup>	<sup>↵</sup>	Call the method Thread.start() to execute the content of the run() method in a dedicated thread. <sup>↵</sup>
DVM5 <sup>↵</sup>	71 <sup>↵</sup>	137 <sup>↵</sup>	Add a default case to this switch <sup>↵</sup>
	53 <sup>↵</sup>	<sup>↵</sup>	Call the method Thread.start() to execute the content of the run() method in a dedicated thread. <sup>↵</sup>
DVM6 <sup>↵</sup>	71 <sup>↵</sup>	139 <sup>↵</sup>	Add a default case to this switch <sup>↵</sup>
	53 <sup>↵</sup>	<sup>↵</sup>	Call the method Thread.start() to execute the content of the run() method in a dedicated thread. <sup>↵</sup>

DVM7 <sup>↵</sup>	71 <sup>↵</sup>	131 <sup>↵</sup>	Add a default case to this switch <sup>↵</sup>
	53 <sup>↵</sup>	<sup>↵</sup>	Call the method Thread.start() to execute the content of the run() method in a dedicated thread. <sup>↵</sup>
DVM8 <sup>↵</sup>	71 <sup>↵</sup>	136 <sup>↵</sup>	Add a default case to this switch <sup>↵</sup>
	53 <sup>↵</sup>	<sup>↵</sup>	Call the method Thread.start() to execute the content of the run() method in a dedicated thread. <sup>↵</sup>
CodePayment <sup>↵</sup>	75 <sup>↵</sup>	130 <sup>↵</sup>	Add a nested comment explaining why ths method is empty, throw an UnsupportedOperationException or complete the implementation <sup>↵</sup>
Network <sup>↵</sup>	44 <sup>↵</sup>	132 <sup>↵</sup>	Use try-with-resources or close this "Socket" in a "finally" clause. <sup>↵</sup>
	73 <sup>↵</sup>	138 <sup>↵</sup>	Use try-with resources or close this "Socket" in a "finally" clause. <sup>↵</sup>
	7 <sup>↵</sup>	141 <sup>↵</sup>	Don't extend "Thread", since the "run" method is not overridden. <sup>↵</sup>
	126 <sup>↵</sup>	146 <sup>↵</sup>	Use try-with resources or close this "Socket" in a "finally" clause. <sup>↵</sup>
MyFrame <sup>↵</sup>	15 <sup>↵</sup>	144 <sup>↵</sup>	Make "labelList" private or transient. <sup>↵</sup>
	168 <sup>↵</sup>	147 <sup>↵</sup>	Refactor this method to reduce its Cognitive Complexity from 64 to the 15 allowed. <sup>↵</sup>
StubTest <sup>↵</sup>	22 <sup>↵</sup>	148 <sup>↵</sup>	Add an end condition to this loop <sup>↵</sup>
	31 <sup>↵</sup>	149 <sup>↵</sup>	Add an end condition to this loop <sup>↵</sup>
	37 <sup>↵</sup>	158 <sup>↵</sup>	Add a default case to this switch <sup>↵</sup>
	66 <sup>↵</sup>	160 <sup>↵</sup>	Add a default case to this switch <sup>↵</sup>
OtherDVMs <sup>↵</sup>	4 <sup>↵</sup>	157 <sup>↵</sup>	Don't extend "Thread", since the "run" method is not overridden. <sup>↵</sup>

- Switch문에 default case 추가 필요
- Thread.run() 직접호출 금지
- Thread extend 금지
- 루프에 end condition 추가

# Quality - 2. FindBugs



- Bugs : 21
- Code smell : 13
- Vulnerability : 0



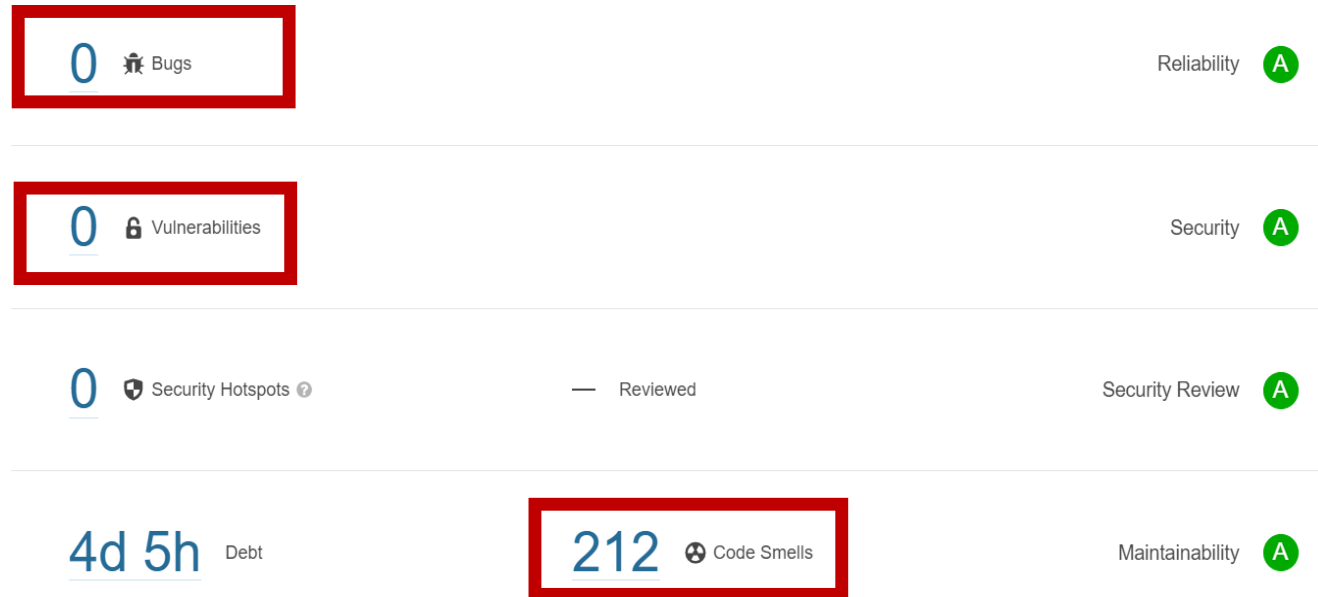
# Quality - 2. FindBugs

File	Line	Redmine issue#	Review
DVM1	75	185	Switch statement found in DVM1.run() where default case is missing
	54	184	DVM1.main(String[]) explicitly invokes run on a thread(did you mean to start it instead?)
	17	183	Unread field: DVM1.STUB_TEST_ID; should this field be static?
DVM2	71	188	Switch statement found in DVM2.run() where default case is missing
	53	187	DVM1.main(String[]) explicitly invokes run on a thread(did you mean to start it instead?)
	17	186	Unread field: DVM1.STUB_TEST_ID; should this field be static?
DVM3	71	191	Switch statement found in DVM2.run() where default case is missing
	53	190	DVM1.main(String[]) explicitly invokes run on a thread(did you mean to start it instead?)
	17	189	Unread field: DVM1.STUB_TEST_ID; should this field be static?
DVM4	71	194	Switch statement found in DVM2.run() where default case is missing
	53	193	DVM1.main(String[]) explicitly invokes run on a thread(did you mean to start it instead?)
	17	192	Unread field: DVM1.STUB_TEST_ID; should this field be static?
DVM5	71	197	Switch statement found in DVM2.run() where default case is missing
	53	196	DVM1.main(String[]) explicitly invokes run on a thread(did you mean to start it instead?)

DVM6	71	200	Switch statement found in DVM2.run() where default case is missing
	53	199	DVM1.main(String[]) explicitly invokes run on a thread(did you mean to start it instead?)
	17	198	Unread field: DVM1.STUB_TEST_ID; should this field be static?
DVM7	71	203	Switch statement found in DVM2.run() where default case is missing
	53	202	DVM1.main(String[]) explicitly invokes run on a thread(did you mean to start it instead?)
	17	201	Unread field: DVM1.STUB_TEST_ID; should this field be static?
DVM8	71	206	Switch statement found in DVM2.run() where default case is missing
	53	205	DVM1.main(String[]) explicitly invokes run on a thread(did you mean to start it instead?)
	17	203	Unread field: DVM1.STUB_TEST_ID; should this field be static?
CardPayment	1	178	Unused field: CardPayment.card_info
	11	179	Unused field: CardPayment.isPrePayment
	30	180	.CardPayment.generateCode(Drink) uses the nextDouble method of Random to generate a random integer; using nextInt is more efficient
MyFrame	176	173	Dead store to aa in MyFrame\$PadInput.actionPerformed(ActionEvent)
	201	174	Switch statement found in MyFrame\$PadInput.actionPerformed(ActionEvent) where default case is missing
StubTest	11	175	Found reliance on default encoding in StubTest.main(String[]):new java.util.Scanner(InputStream)
	37	176	Switch statement found in StubTest.test(int, int,int,String) where default case is missing.
Controllers	1	181	Unused field: Controller.card_info
	84	182	Integer is incompatible with expected argument type Code in Controller.insertCard(int, boolean)
Teset2	4	211	Dead store to m in Test2.main(String[])

- Switch문에 default case추가 필요
- Thread.run() 대신 Thread.start() 사용
- 사용하지 않은 field
- nextDouble보다 nextInt사용 요구

# Quality - 3.PMD



- Bugs : 0
- Code smell : 212
- Vulnerability : 0

-> 전체 212개의 버그 및 code smell 중 19개의 유의미한 문제점 추려냄

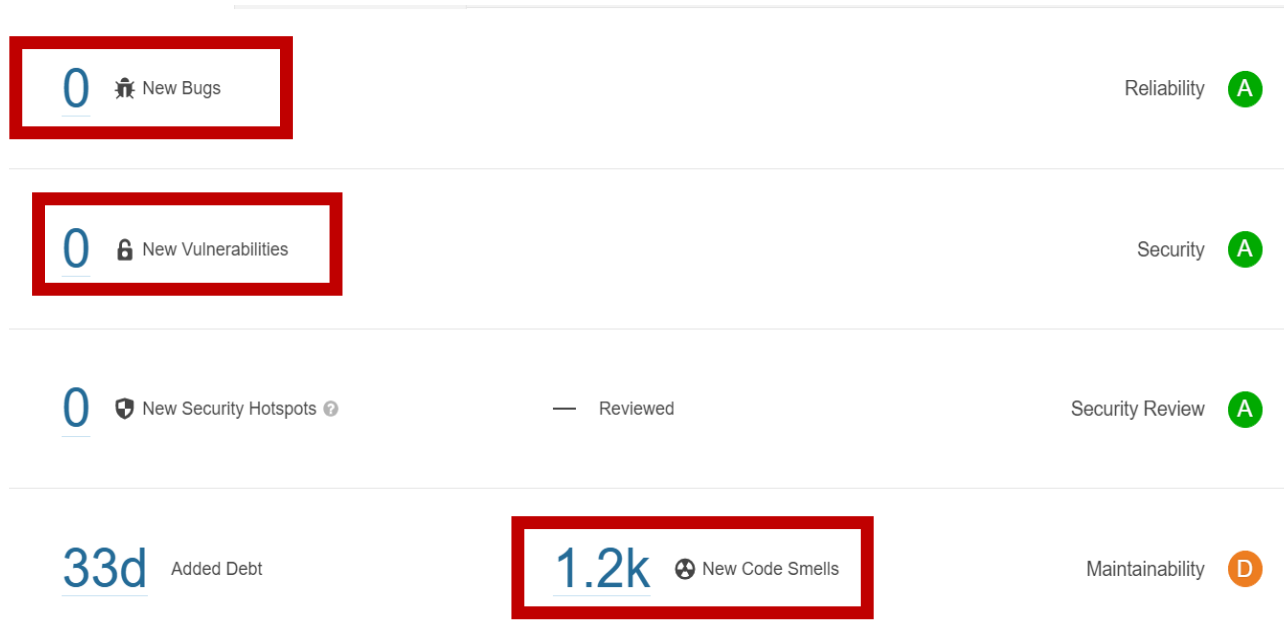
# Quality - 3.PMD

File	Line	Redmine issue#	Review
DVM	7	217	Avoid using implementation types like 'ArrayList'; use the interface instead.
	24	214	All classes, Interfaces, enums, and annotations must belong to a named package
StubTest	8	215	Avoid using final local variables, turn them into fields
	51	219	Switch statements should have a default label.
	53	220	Avoid catching generic exception such as NullPointerException, Exception in try-catch block
	80	223	Switch statements should have a default label.
	82	224	Avoid catching generic exceptions such as NullPointerException, RuntimeException, Exception in try-catch block
	86	218	The method 'test[int,int,int,String]' has a cyclomatic complexity of 17.
	87	212	Each class should declare at least one constructor

Card	2	216	Private field 'card_num' could be made final; it is only initialized in the declaration or constructor.
	19	213	All classes, Interfaces, enums and annotations must belong to a named package
Code	4	222	Private field 'drink' could be made final; it is only initialized in the declaration or constructor.
CodePayment	7	226	Avoid unnecessary constructors - the compiler will generate these for you
	7	229	Document empty constructor.
	23	230	Consider simply returning the value vs storing it in local variable 'drink_info'
Message	55	227	Classes implementing Serializable should set a serialVersionUID.
	55	228	Each class should declare at least one constructor.
Drink	6	221	Private field 'name' could be made final; It is only initialized in the declaration or constructor.
	9	225	Private field 'imgURL' could be made final; it is only initialized in the declaration or constructor.

- Implementation type 사용하기 보다는 interface 사용
- Final local 변수 사용 피하기
- Switch문에 default case 추가
- Private 필드는 final로 선언 필요

# Quality - 4. CheckStyle



- Bugs : 0
- Code smell : 1.2k
- Vulnerability : 0

-> 전체 1.2k개의 code smell 중 88개의 유의미한 문제점 추려냄

-> 동일한 rule을 위반한 code smell로 묶어 5개의 이슈로 redmine에 발행

# Quality - 4. CheckStyle

\* 중복해서 발생한 rule은 묶어서 표로 작성

Rule <sup>↗</sup>	Review <sup>↗</sup>	Class-Line <sup>↗</sup>	Redmine issue # <sup>↗</sup>
Avoid Star Import <sup>↗</sup>	Using the '*' form of import should be avoided - javax.swing.*. <sup>↗</sup>	MyFrame.java - L1 <sup>↗</sup> MyFrame.java - L2 <sup>↗</sup>	231 <sup>↗</sup>
Nested If Depth <sup>↗</sup>	Nested if-else depth is 2 (max allowed is 1) <sup>↗</sup>	MyFrame.java - L177 <sup>↗</sup>	232 <sup>↗</sup>
Visibility Modifier <sup>↗</sup>	Variable must be private and have accessor methods. <sup>↗</sup>	CardPayment.java - L9 <sup>↗</sup> Controller.java - L4, L5, L6, L7, L8, L9, L10, L11 <sup>↗</sup> DVM1.java - L13, L14, L15, L16 <sup>↗</sup> DVM2.java - L13, L14, L15, L16 <sup>↗</sup> DVM3.java - L13, L14, L15, L16 <sup>↗</sup> DVM4.java - L13, L14, L15, L16 <sup>↗</sup> DVM5.java - L13, L14, L15, L16 <sup>↗</sup> DVM6.java - L13, L14, L15, L16 <sup>↗</sup> DVM7.java - L13, L14, L15, L16 <sup>↗</sup> DVM8.java - L13, L14, L15, L16 <sup>↗</sup> MyFrame.java - L9, L10, L11, L12, L13, L14, L15, , L18, L19, L20, L21, L22, L25, L26, L29 <sup>↗</sup> Network.java - L8 <sup>↗</sup> OtherDVMs.java - L5, L6 <sup>↗</sup>	233 <sup>↗</sup>


Unused Imports <sup>↗</sup>	Unused Imports <sup>↗</sup> (java.util.Arrays, java.io.IOException, java.util.ArrayList,java.util.List, java.net.ServerSocket,java.util.Arrays) <sup>↗</sup>	CardPayment.java - L2 <sup>↗</sup> DVM1.java - L1 <sup>↗</sup> DVM2.java - L1 <sup>↗</sup> DVM3.java - L1 <sup>↗</sup> DVM4.java - L1 <sup>↗</sup> DVM5.java - L1 <sup>↗</sup> DVM6.java - L1 <sup>↗</sup> DVM7.java - L1 <sup>↗</sup> DVM8.java - L1 <sup>↗</sup> Drink.java - L1, L2 <sup>↗</sup> Network.java - L3 <sup>↗</sup> OtherDVMs.java - L2 <sup>↗</sup>	234 <sup>↗</sup>
Illegal Catch <sup>↗</sup>	Catching 'Exception' is not allowed. <sup>↗</sup>	DVM1.java - L138 <sup>↗</sup> DVM2.java - L84, L134, L153, L179 <sup>↗</sup> DVM3.java - L84, L134, L153, L179 <sup>↗</sup> DVM4.java - L84, L134, L153, L179 <sup>↗</sup> DVM5.java - L84, L134, L153, L179 <sup>↗</sup> DVM6.java - L84, L134, L153, L179 <sup>↗</sup> DVM7.java - L84, L134, L153, L179 <sup>↗</sup> DVM8.java - L84, L134, L153, L179 <sup>↗</sup> Network.java - L60, L91, L115, L141 <sup>↗</sup> StubTest.java - L53, L82 <sup>↗</sup>	236 <sup>↗</sup>

- import 시 \* 사용 피하기
- If문 중첩 max 1
- 사용하지 않은 라이브러리 제거 필요

<input type="checkbox"/>	#	유형	상태	우선순위	제목	담당자	변경	
<input type="checkbox"/>	237	결함	신규	보통	[2nd Cycle] [BFT] 음료 구매 완료 메시지 출력 확인		2021/05/30 20:41	...
<input type="checkbox"/>	236	결함	신규	보통	[2nd Cycle] [Check Style] Iligal Catch		2021/05/28 19:45	...
<input type="checkbox"/>	234	결함	신규	보통	[2nd Cycle] [Check Style] Unused Imports		2021/05/28 19:39	...
<input type="checkbox"/>	233	결함	신규	보통	[2nd Cycle] [Check Style] Visibility Modifier		2021/05/28 19:35	...
<input type="checkbox"/>	232	결함	신규	보통	[2nd Cycle] [Check Style] Nested If Depth		2021/05/28 19:47	...
<input type="checkbox"/>	231	결함	신규	보통	[2nd Cycle] [Check Style] Avoid Star Import		2021/05/28 19:41	...
<input type="checkbox"/>	230	결함	신규	보통	[2nd Cycle] [PMD] CodePayment.java - L23		2021/05/28 19:11	...
<input type="checkbox"/>	229	결함	신규	보통	[2nd Cycle] [PMD] CodePayment.java - L7 - 2		2021/05/28 19:10	...
<input type="checkbox"/>	228	결함	신규	보통	[2nd Cycle] [PMD] Message.java - L55 - 2		2021/05/28 19:01	...
<input type="checkbox"/>	227	결함	신규	보통	[2nd Cycle] [PMD] Message.java - L55		2021/05/28 18:59	...
<input type="checkbox"/>	226	결함	신규	보통	[2nd Cycle] [PMD] CodePayment.java - L7		2021/05/28 18:59	...
<input type="checkbox"/>	225	결함	신규	보통	[2nd Cycle] [PMD] Drink.java - L9		2021/05/28 18:58	...
<input type="checkbox"/>	224	결함	신규	보통	[2nd Cycle] [PMD] StubTest - L82		2021/05/28 18:57	...
<input type="checkbox"/>	223	결함	신규	보통	[2nd Cycle] [PMD] StubTest - L80		2021/05/28 18:57	...
<input type="checkbox"/>	222	결함	신규	보통	[2nd Cycle][PMD] Code.java - L4		2021/05/28 18:56	...
<input type="checkbox"/>	221	결함	신규	보통	[2nd Cycle] [PMD] Drink.java - L6		2021/05/28 18:56	...
<input type="checkbox"/>	220	결함	신규	보통	[2nd Cycle] [PMD] StubTest - L53		2021/05/28 18:56	...
<input type="checkbox"/>	219	결함	신규	보통	[2nd Cycle] [PMD] StubTest - L51		2021/05/28 18:56	...
<input type="checkbox"/>	218	결함	신규	보통	[2nd Cycle] [PMD] StubTest - L86		2021/05/28 18:55	...
<input type="checkbox"/>	217	결함	신규	보통	[2nd Cycle] [PMD] DVM.java - L7		2021/05/28 18:55	...
<input type="checkbox"/>	216	결함	신규	보통	[2nd Cycle][PMD] Card.java - L2		2021/05/28 18:55	...
<input type="checkbox"/>	215	결함	신규	보통	[2nd Cycle] [PMD] StubTest - L8		2021/05/28 18:54	...
<input type="checkbox"/>	214	결함	신규	보통	[2nd Cycle] [PMD] DVM.java - L24		2021/05/28 18:54	...
<input type="checkbox"/>	213	결함	신규	보통	[2nd Cycle] [PMD] Card.java - L19		2021/05/28 18:53	...
<input type="checkbox"/>	212	결함	신규	보통	[2nd Cycle] [PMD] StubTest - L87		2021/05/28 18:53	...

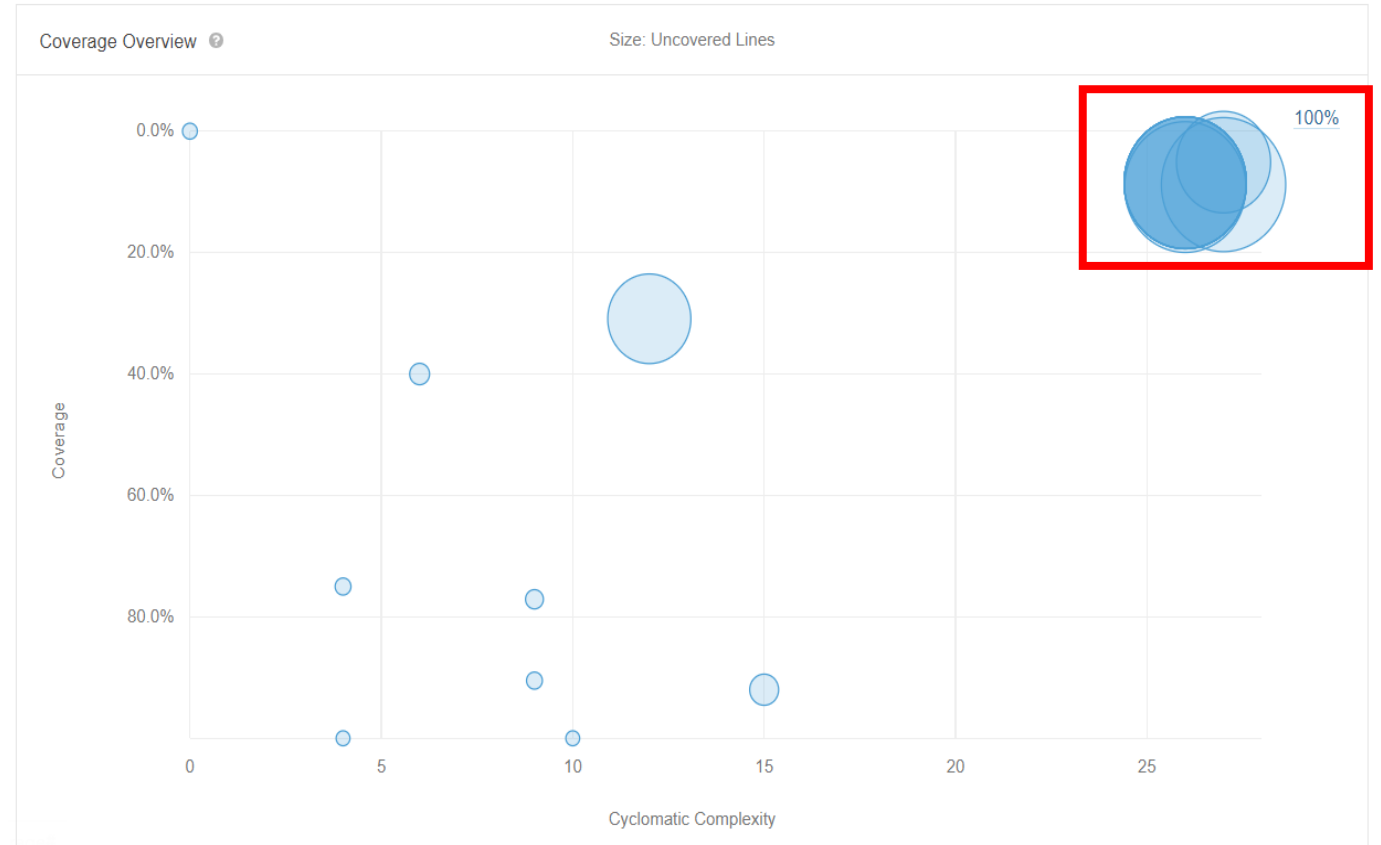
- Redmine에 결함으로 등록해 놓음

# Quality - 5. Coverage

Coverage		COVERAGE ON NEW CO...	
Overview 			
On new code			
Coverage	26.4%		
Lines to Cover	1,482		
Uncovered Lines	1,044		
Line Coverage	29.6%		
Conditions to Cover	244		
Uncovered Conditions	226		
Condition Coverage	7.4%		

- 전체 코드에 대한 coverage = 26.4%
- Line에 대한 coverage = 29.6%
- Condition coverage = 7.4%

**➔ Unit test case가 상당히 부족, 부실**



# Quality - 5. Coverage

	Coverage on New Code	Uncovered Lines on New Code	Uncovered Conditions on New Code
Card.java	100%	0	0
CardPayment.java	77.1%	4	4
Code.java	75.0%	2	0
CodePayment.java	40.0%	6	0
Controller.java	5.1%	80	32
Drink.java	90.5%	2	0
DVM.java	-	0	0
DVM1.java	8.8%	110	24
DVM2.java	9.2%	107	22
DVM3.java	8.5%	108	22
DVM4.java	8.5%	108	22
DVM5.java	8.5%	108	22
DVM6.java	8.5%	108	22
DVM7.java	8.5%	108	22
DVM8.java	8.5%	108	22
Message.java	100%	0	0
MsgType.java	0.0%	1	0
MyFrame.java	-	0	0
Network.java	30.9%	69	7
OtherDVMs.java	92.0%	15	5
StubTest.java	-	0	0
Test2.java	-	0	0

- DVM1.java~DVM8.java  
→ 실제 DVM1~8의 함수를 실행하는 test code가 없어서 한 자리 수 coverage를 보여줌.
- Controller.java  
→ Controller의 주요 기능을 테스트하는 unit test code가 @Disabled처리 되어 있어서 coverage가 작게 나온 것 확인.



# Quality - 5. Coverage

	Line Coverage on New Code	Uncovered Lines on New Code
Card.java	100%	0
CardPayment.java	85.2%	4
Code.java	75.0%	2
CodePayment.java	40.0%	6
Controller.java	7.0%	80
Drink.java	90.5%	2
DVM.java	-	0
DVM1.java	10.6%	110
DVM2.java	10.8%	107
DVM3.java	10.0%	108
DVM4.java	10.0%	108
DVM5.java	10.0%	108
DVM6.java	10.0%	108
DVM7.java	10.0%	108
DVM8.java	10.0%	108
Message.java	100%	0
MsgType.java	0.0%	1
MyFrame.java	-	0
Network.java	29.6%	69
OtherDVMs.java	93.7%	15
StubTest.java	-	0
Test2.java	-	0

- 각 클래스별 line coverage = 29.6%

Card.java	-	0
CardPayment.java	50.0%	4
Code.java	-	0
CodePayment.java	-	0
Controller.java	0.0%	32
Drink.java	-	0
DVM.java	-	0
DVM1.java	0.0%	24
DVM2.java	0.0%	22
DVM3.java	0.0%	22
DVM4.java	0.0%	22
DVM5.java	0.0%	22
DVM6.java	0.0%	22
DVM7.java	0.0%	22
DVM8.java	0.0%	22
Message.java	-	0
MsgType.java	-	0
MyFrame.java	-	0
Network.java	41.7%	7
OtherDVMs.java	64.3%	5
StubTest.java	-	0
Test2.java	-	0

- 각 클래스별 condition coverage = 7.4%

# Quality - 6. Cyclomatic Complexity

복잡도 V(G)	위험성 평가
1-10	단순 프로그램, 위험성 낮음
11-20	다소 복잡한 프로그램, 위험성 중간
21-50	복잡한 프로그램, 위험성 높음
51 이상	테스트 불가능한 프로그램, 위험성 매우 높음

Cyclomatic Complexity 390

Card.java	4
CardPayment.java	9
Code.java	4
CodePayment.java	6
Controller.java	27
Drink.java	9
DVM.java	0
DVM1.java	27
DVM2.java	26
DVM3.java	26
DVM4.java	26

DVM5.java	26
DVM6.java	26
DVM7.java	26
DVM8.java	26
Message.java	10
MsgType.java	0
MyFrame.java	67
Network.java	12
OtherDVMs.java	15
StubTest.java	17
Test2.java	1

코드를 직접 본 결과 if문의  
중첩이 다수 발견됨



**다소 높은 Cyclomatic  
Complexity 발생**

# PART 5

# Overall Review

# Overall Review

1

## Spec review

- reference(ref) 를 usecase 에 따라 수정 필요

2

## System testing

- 마지막 음료 구매 후 나오는 메시지에서 DVM번호가 하나씩 밀려서 나오는 문제 수정 필요.

3

## Static Code Analysis

- 각 DVM을 객체가 아닌 클래스로 만들었기 때문에 중복코드가 많고 그만큼 coverage도 낮았음. 또한, coverage 향상을 위해 unit test code 수정 필요.

4

## Communication

-금요일에 받은 완성된 코드에서 오류가 있어서 수정 요구 후 오후에 다시 받음, 소통은 원활하게 잘 됨.

**감사합니다.**