

# 2021 Software V&V

Static Analysis & 2<sup>nd</sup> System Testing Report

[OOAD Team A2] & [V&V Team 2]

201711431 차희수

201611218 이기현

201811297 주윤지

201510621 김민혁

## Contents

# 1. System testing

- 1.1 Category Partition Test Case
- 1.2 Generate Test Case Using TSL Generator
- 1.3 Category Partition Testing Result
- 1.4 Brute-force Testing
- 1.5 TestLink / Redmine
- 1.6 Summary

# 2. Static Analysis

- 2.1 Static Analysis - SonarQube [ Bugs, Security Hotspots, CodeSmells]
- 2.2 Redmine 일감처리
- 2.3 Summary
- 2.4 Code Coverage - Jacoco

# 1 System Testing

- 1.1 Category Partition Test Case

※ 지난 test 의 test result 와의 비교를 위해 test case 는 그대로 사용

Group	Category	Value
Mode	Mode	Payment
		PrePayment
		Input Authentication Code
		Admin
Verification	Card Number Length	==12
		12>    12<
	Balance	True
		False
	Exist in CardList	True
		False
	Admin Code Length	==6
		6>    6<
	Admin Code Consistency	True
		False
	User Auth Code Length	==4
		>4    <4
	User Auth Code Consistency	True
		False
	VM Address Length	1~10
		>10    <1
	VM Address Consistency	True
		False
	Stock	True
		False
		Other DVM True
		Other DVM False
	VMList	True
		False

Input	Menu	1~20
		Else
	Cancel	True
		False
Function	Payment	Balance Change
		Not Change
	Prepay	Balance Change
		Not Change
	Stock change	Change
		Not Change
	Refund	Balance Change
		Not Change
	Switch Mode	True
		False
	Create Auth Code	True
		False
	Authentication code duplicated or not	True
		False
Output	Showing Menu	True
		False
	Showing VMList	True
		False
	Showing Payment	True
		False
	Showing Authentication Code Screen:	True
		False
	Showing Admin Mode Screen	True
		False
	Showing Error Message	True
		False

Network	Vending machine startup request and address	Request Fail
		Request Success but not Msg type 1.
		Request Success and Msg type1
	Vending Machine Initiated Response and Address	Response Fail
		Response Success but not Msg type 2
		Response Success and Msg type2
	Shut Down Vending Machine	Send Fail.
		Send Success but not Msg type 3.
		Send Success and Msg type3
	Stock Status Request	Request Fail
		Request Success but not Msg type 4
		Request Success and Msg type4
	Stock Status Response	Response Fail
		Response Success but not Msg type 5
		Response Success and Msg type5
	Authentication Code Generation Request	Request Fail.
		Request Success but not Msg type 6
		Request Success and Msg type6
	Authentication Code Response	Response Fail
		Response Success but not Msg type 7
		Response Success and Msg type7
	Refund request	Request Fail.
		Request Success but not Msg type 8
		Request Success and Msg type8
	Refund Request Response	Response Fail
		Response Success but not Msg type 9
		Response Success and Msg type9

## 1.2 Generate Test Case Using TSL Generator

- TSL Generator에 넣을 문서 작성

---

#Mode		
Mode:	Payment. PrePayment. Input Authentication Code. Admin	[property PayMode] [property PrePayMode] [property AuthMode] [property AdminMode]
#Verification		
Card Number Length:		
	==12. 12>    12<.	[property CardNumberLength][if PayMode    PrePayMode] [single]
Remain Money:		
	Exit. No.	[if CardNumberLength][property RemainMoney] [single]
Exist in cardlist:		
	Yes. No.	[if RemainMoney][property ExistInCardlist] [single]
Admin Mode Code Length:		
	==6. 6>    6<.	[if StartMode    OffMode][property AdminCodeLength] [single]
Admin Mode Code Consistency:		
	Yes. No.	[if AdminCodeLength][property AdminCodeConsistency] [single]
User authentication code:		
	==4. >4    <4.	[if AuthMode][property UserCodeLength] [single]
User authentication Code Consistency:		
	Yes. No.	[if UserCodeLength][property UsercodeConsistency] [if UserCodeLength][single]
Vending Machine Address:		
	1~10. >10    <1.	[if AdminCodeConsistency][property VMAddressLength] [single]
Vending Machine Address Consistency :		
	Yes. No.	[if VMAddressLength][property VMAddressConsistency] [error]
Stock :		
	True. False.	[property Stock]

---

- TSL Generator Output

Test Case 1                      <single>  
Card Number Length : 12> || 12<

Test Case 2                      <error>  
Card Number Length : N/A

Test Case 3                      <single>  
Remain Money : No

Test Case 4                      <single>  
Exist in cardlist : No

Test Case 5                      <single>  
Admin Mode Code Length : 6> || 6<

Test Case 6                      <error>  
Admin Mode Code Length : N/A

Test Case 7                      <single>  
Admin Mode Code Consistency : No

Test Case 8                      <single>  
User authentication code : >4 || <4

Test Case 9                      <error>  
User authentication code : N/A

Test Case 10                      <single> (follows [if])  
User authentication Code Consistency : No

Test Case 11                      <single>  
Vending Machine Address : >10 || <1

### 1.3 Category Partition Testing

※ 지난 test 의 test result 와의 비교를 위해 test case 는 그대로 사용

Num	Description	Pass/Fail	Fail-> Pass 이유
1	입력카드 번호가 >12    <12인 경우 오류 메시지 출력	P	
2	결제 카드의 잔액이 주문 음료보다 적은경우 오류메시지 출력	P	
3	입력 카드가 리스트에 존재하지 않은 경우 오류메시지 출력	P	
4	관리자 코드가 >6    <6인 경우 오류 메시지 출력	P	
5	올바르지 않은 관리자 코드 입력시 오류메시지 출력	P	
6	사용자 인증 코드가 >4    <4인경우 오류 메시지 출력	P	
7	올바르지 않은 사용자 코드 입력시 오류메시지 출력	P	
8	주소가 프로그램 상에서 형식에 맞게 입력	P	
9	취소버튼 클릭시 메뉴 선택 대기 화 면으로 전환	P	
10	올바른 관리자 코드 입력시 관리자모 드로 화면전환	P	
11	관리자 모드와 유저 모드의 구분이 모호하여 테스트 불가능	F-> P	SRS (V4) 16p 3.2.6 Admin Mode 관리자 모드와 사용자 모드에 대 해 구체적으로 명시되어 수정된 것을 확인.
12	오류 상황시 오류 메시지 화면에 출 력	P	
13	DVM 시작시 Request Success and Msg type1 확인	P	
14	DVM 초기화시 Respond Success and Msg type2 확인	P	
15	DVM 종료시 Send Success and Msg type3 확인	P	

16	VMList 상의 다른 DVM에게 재고 메시지 요청시 Request Success and Msg type4 확인	P	
17	재고 확인 요청 받은 경우 Respond Success and Msg type5 전송 확인	P	
18	다른 DVM에게 선결제시 Request Success and Msg type6 전송 확인	P	
19	선결제 완료시 Respond Success and Msg type7 전송 확인	P	
20	환불 요청시 Request Success and Msg type 8 전송 확인	F->P	환불 과정에 대한 이해가 달라서 오류라고 판단 -> 개발팀과의 회의결과 선결제에서만 환불진행이 된다고 이해 후 수정
21	환불 완료시 Respond Success and Msg type 9 전송 확인	P	
22	현 DVM내 재고가 존재하는 음료 선택 후 결제 성공시 다시 메뉴 대기 화면으로 전환 확인	P	
23	현 DVM내 재고가 존재하는 음료를 메뉴 대기 화면에서 선택시 결제 화면으로 전환	P	
24	연결된 다른 DVM이 존재하지 않은 상황에 재고가 없는 음료 선택시 오류메시지 테스트 불가능	F->P	개발팀과의 오류 문구 추가 협의 ->오류 문구 작동 확인 완료
25	현 DVM내 재고가 존재하는 음료를 메뉴 대기 화면에서 확인 가능(초록색)	P	
26	재고가 있고 다른 자판기가 꺼져 있을때 선결제 실패 확인	P	
27	재고가 있고 다른 자판기가 꺼져 있으며 결제 화면이 나타났을때 선결제 실패 확인	P	
28	재고가 없고 다른 자판기가 켜져 있을 때 선결제 화면 출력 확인	P	
29	재고가 없고 다른 자판기가 꺼져 있		콘솔창을 통해 재고/잔고 가시적

30	<p>을 때 선결제 실패 확인</p> <p>다른 자판기에 재고가 있고 켜져있을 때 잔고와 재고가 정상적으로 차감되는 선결제 성공 확인</p>	<p>F-&gt;P</p> <p>F-&gt;P</p>	<p>확인 완료</p> <p>콘솔창을 통해 재고/잔고 가시적 확인 완료</p>
31	<p>다른 자판기에 재고가 있고 켜져있을 때 인증코드가 중복없이 생성되며 잔고와 재고가 정상적으로 차감되는 선결제 성공하는 것 확인</p>	F->P	<p>콘솔창을 통해 재고/잔고 가시적 확인 완료</p>
32	<p>다른 자판기에 재고가 있으나 꺼져있을 때 잔고와 재고가 정상적으로 차감되며 선결제 실패 확인</p>	F->P	<p>콘솔창을 통해 재고/잔고 가시적 확인 완료</p>
33	<p>다른 자판기에 재고가 있으나 꺼져있을 때 잔고와 재고가 정상적으로 차감되며 선결제 화면 전환 실패 확인</p>	P	
34	<p>다른 자판기에 재고가 없고 켜져있을 때 선결제 실패 확인</p>	P	
35	<p>다른 자판기에 재고가 없고 꺼져있을 때 선결제 실패 확인</p>	F->P	<p>인증번호가 0으로 시작할 경우 오류 해결 확인 완료</p>
36	<p>재고가 있고 다른 자판기가 켜져있을 때 인증코드 입력을 통한 구매 성공 확인</p>	F->P	<p>인증번호가 0으로 시작할 경우 오류 해결 확인 완료</p>
37	<p>재고가 있고 다른 자판기가 꺼져있을 때 인증코드 입력을 통한 구매 성공 확인</p>	F->P	<p>인증번호가 0으로 시작할 경우 오류 해결 확인 완료</p>
38	<p>재고가 없고 다른 자판기가 켜져있으며 재고가 있을 때 인증코드 입력을 통한 구매 성공 확인</p>	F->P	<p>인증번호가 0으로 시작할 경우 오류 해결 확인 완료</p>
39	<p>재고가 없고 다른 자판기가 꺼져있으며 재고가 있을 때 인증코드 입력을 통한 구매 성공 확인</p>	F->P	<p>인증번호가 0으로 시작할 경우 오류 해결 확인 완료</p>
40	<p>재고가 없고 다른 자판기가 켜져있으며 재고가 없을 때 인증코드 입력을 통한 구매 성공 확인</p>	F->P	<p>인증번호가 0으로 시작할 경우 오류 해결 확인 완료</p>
41	<p>재고가 없고 다른 자판기가 꺼져있으며 재고가 있을 때 인증코드 입력을</p>	F->P	<p>인증번호가 0으로 시작할 경우 오류 해결 확인 완료</p>

42	통한 구매 성공 확인 재고가 없고 다른 자판기가 꺼져있으며 재고가 없을 때 인증코드 입력을 통한 구매 성공 확인	F->P	인증번호가 0으로 시작할 경우 오류 해결 확인 완료
43	재고가 없고 다른 자판기가 켜져있으며 재고가 없을 때 인증코드 입력을 통한 구매 성공 확인	F->P	콘솔창을 통해 재고/잔고 가시적 확인 완료
44	선결제를 통해 구매 과정 중 다른 자판기에서 결제를 통해 재고가 0이 되었을 경우 환불이 실행되며 잔고가 복구가 되는 지 확인	P	
45	자판기가 정상적으로 실행되는지 확인	P	
46	자판기가 정상적으로 종료되는지 확인	F->P	콘솔창을 통해 재고/잔고 가시적 확인 완료
47	재고가 있고 다른 자판기가 켜져있을 때 재고 복구 성공 확인	F->P	콘솔창을 통해 재고/잔고 가시적 확인 완료
48	재고가 없고 다른 자판기가 꺼져있을 때 재고 복구 성공 확인	F->P	콘솔창을 통해 재고/잔고 가시적 확인 완료
49	재고가 있고 다른 자판기가 꺼져있을 때 재고 복구 성공 확인	F->P	콘솔창을 통해 재고/잔고 가시적 확인 완료
50	재고가 없고 다른 자판기가 켜져있을 때 재고 복구 성공 확인	F->P	콘솔창을 통해 재고/잔고 가시적 확인 완료
51	다른 자판기에 재고가 있고 다른 자판기가 켜져있을 때 재고 복구 성공 확인	F->P	콘솔창을 통해 재고/잔고 가시적 확인 완료
52	다른 자판기에 재고가 있고 다른 자판기가 꺼져있을 때 재고 복구 성공 확인	F->P	콘솔창을 통해 재고/잔고 가시적 확인 완료
53	다른 자판기에 재고가 없고 다른 자판기가 켜져있을 때 재고 복구 성공 확인	F->P	콘솔창을 통해 재고/잔고 가시적 확인 완료
54	다른 자판기에 재고가 없고 다른 자판기가 꺼져있을 때 재고 복구 성공 확인	P	

- 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**

## 1.4 Brute-force Testing

※ 지난 test 의 test result 와의 비교를 위해 test case 는 그대로 사용

Num	Test	Description	Pass/Fail
1	카드 결제 기능	카드번호(12자리미만)으로 입력된 경우 오류 문구가 출력 되는지 확인한다.	P
2	카드 결제 기능	잔액이 없는 카드로 결제가 가능한지 확인한다.	P
3	카드 결제 기능	정상 결제된 경우 잔액이 메뉴의 가격만큼 차감되었는지 확인한다.	P
4	카드 결제 기능	정상 결제된 경우 음료의 재고가 차감되었는지 확인한다.	P
5	카드 결제 기능	유효하지 않은 카드로 결제가 가능한지 확인한다.	P
6	카드 결제 기능(선결제)	잔액이 없는 카드로 결제가 가능한지 확인한다.	P
7	카드 결제 기능(선결제)	유효하지 않은 카드로 결제가 가능한지 확인한다.	P
8	카드 결제 기능(선결제)	정상 결제된 경우 잔액이 메뉴의 가격만큼 차감되었는지 확인한다.	P
9	재고 관리 기능	실제 재고와 측정된 재고 값을 비교한다.	P
10	관리자 모드 진입 기능	올바른 관리자 코드를 입력했을 때 관리자 모드로 전환하는지 확인한다.	P
11	관리자 모드 진입 기능	올바르지 않은 관리자 코드를 입력했을 때 오류 문구가 출력되고 메뉴 선택 대기 상태로 전환하는지 확인한다.	P
12	메뉴 선택 기능	재고가 있는 메뉴를 선택했을 때 카드 결제 대기 상태로 전환하는지 확인한다	P
13	메뉴 선택 기능	재고가 없는 메뉴를 선택했을 때 다른 자판기로 재고 여부 메시지가 전송되는지 확인하는지 확인한다.	P
14	인증코드 입력 진입 기능	인증코드 입력 대기 상태로 전환되는지 확인한다.	P
15	인증코드 입력 기능	유효한 인증코드를 입력한 경우 해당 음료가 제공되는지 확인한다.	P

16	인증코드 입력 기능	유효하지 않은 인증코드를 입력한 경우 오류 문구가 출력되고 메뉴 선택 대기 상태로 전환하는지 확인한다.	P
17	다른 자판기 선택 기능	자판기를 선택했을 때 Card Payment가 실행되는지 확인한다.	P
18	다른 자판기 선택 기능	VMList에 존재하며 현재 실행 중인 다른 자판기들을 모두 화면에 보여주는지 확인한다.	P
19	다른 자판기 재고 여부 응답 기능	다른 자판기로부터 재고 여부 확인 메시지를 수신한 경우 해당 메뉴의 재고 여부를 메시지로 응답하는지 확인한다.	P
20	위치 설정	입력한 주소가 null일 경우 재입력을 요구하는지 확인한다.	F->P
21	위치 설정	입력한 주소가 null이 아닐 경우 Broadcast로 자판기 시작 메시지가 전송되는지 확인한다.	P
22	자판기 시작 메시지 응답 기능	다른 자판기로부터 자판기 시작 메시지를 수신한 경우 해당 자판기에게 작동중임을 알리는 메시지를 전송하는지 확인한다.	P
23	자판기 시작 메시지 응답 기능	다른 자판기로부터 자판기 시작 메시지를 수신한 경우 VMList에 해당 자판기를 추가하는지 확인한다.	P
24	자판기 종료 기능	자판기 종료를 선택할 경우 VMList의 다른 자판기들에게 자판기 종료 메시지를 전송하는지 확인한다.	F->P
25	자판기 종료 기능	자판기를 종료한 경우 화면 상에 종료된 자판기가 보이지 않는지 확인한다.	P
26	자판기 종료 메시지 응답 기능	다른 자판기로부터 자판기 종료 메시지를 수신한 경우 VMList에서 해당 자판기가 삭제되는지 확인한다.	P
27	작동중임을 알려주는 걸 수신하는 기능	다른 자판기로부터 작동중임을 알리는 메시지를 수신한 경우 해당 자판기가 VMList에서 추가되는지 확인한다.	P
28	재고 여부 수신 기능	다른 자판기로부터 재고 여부 메시지(재고 있음)를 수신한 경우 해당 자판기가 선결제 선택 자판기에 포함되었는지 확인한다.	P
29	재고 여부 수신 기능	다른 자판기로부터 재고 여부 메시지(재고 없	P

		음)를 수신한 경우 해당 자판기가 선결제 선택 자판기에 제외되었는지 확인한다	
30	환불 수신 기능	다른 자판기로부터 환불 요청 메시지를 수신한 경우 결제했던 카드의 잔액이 정상적으로 복구되는지 확인한다.	F->P
31	생성된 인증코드 수신 기능	다른 자판기로부터 생성된 인증코드를 메시지로 수신할 경우 해당 인증코드를 30초간 정상 출력하는지 확인한다.	P
32	인증코드 생성 기능	중복된 인증코드를 생성하지 않으며, 4자리의 올바른 인증코드를 생성하는지 확인한다.	P
33	인증코드 생성 응답 기능	다른 자판기로부터 인증코드 생성 요청 메시지를 수신할 경우 & 선결제 대상 음료의 재고가 1 이상일 경우 인증코드를 생성하여 해당 자판기에게 전송하는지 확인한다.	P
34	인증코드 생성 응답 기능	다른 자판기로부터 인증코드 생성 요청 메시지를 수신할 경우 & 선결제 대상 음료의 재고가 0일 경우 해당 자판기에게 환불 요청 메시지를 전송하는지 확인한다.	P
35	취소 기능	취소 버튼을 눌렀을 때, 메뉴 선택 대기 상태로 전환하는지 확인한다.	P

- 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**

## 1.5 TestLink / Redmine

### [TestLink]

The screenshot shows the TestLink interface for a project named 'Java\_Test\_Project'. At the top, there is a checkbox for 'Show metrics only for active test plans' which is checked. Below this, the 'Test Project Progress' section displays a progress bar at 100% (89/89). A table below the progress bar shows the following metrics: 실행안함: 0% [0/89], 통과: 100% [89/89], 실패: 0% [0/89], 중단: 0% [0/89]. A note explains that project progress is calculated by summing up all executions for a specific execution result for all platforms within all test plans. Below this, the 'Test Plan Progress (ATTENTION: Only Test Plans with ACTIVE BUILDS are considered)' section is visible. A table titled '메트릭 현황' (Metric Status) shows the following data:

테스트 계획	사용하는 테스트 케이스	실행안함 [%]	통과 [%]	실패 [%]	중단 [%]	Progress [%]
DVM Test Plan - Overall Progress: 100%	89	0	100	0	0	100

Additional text at the bottom of the screenshot includes: 'This Report shows the progress for the whole project and all (or only active) test plans defined within project.', 'Overall Progress for test plans is calculated based upon the progress of each platform (if in use).', 'Test Project Progress is calculated based upon the progress of all (or only active) test plans.', and 'TestLink에서 생성 : 2021/06/01 15:11:15'.

-2차 Test 진행 결과 89개 테스트 케이스에 대해 모두 Pass

### [Redmine]

#	유형	상태	우선순위	제목	
<input type="checkbox"/>	38	결함	완료	낮음	CPT-54 : 다른 자판기에 재고가 없고 다른 자판기가 꺼져있을 때 재고 복구 성공 확인
<input type="checkbox"/>	37	결함	완료	낮음	CPT-53 : 다른 자판기에 재고가 없고 다른 자판기가 꺼져있을 때 재고 복구 성공 확인
<input type="checkbox"/>	36	결함	완료	낮음	CPT-52 : 다른 자판기에 재고가 있고 다른 자판기가 꺼져있을 때 재고 복구 성공 확인
<input type="checkbox"/>	35	결함	완료	낮음	CPT-51 : 다른 자판기에 재고가 있고 다른 자판기가 꺼져있을 때 재고 복구 성공 확인
<input type="checkbox"/>	34	결함	완료	낮음	CPT-50 : 재고가 없고 다른 자판기가 꺼져있을 때 재고 복구 성공 확인
<input type="checkbox"/>	33	결함	완료	낮음	CPT-49 : 재고가 있고 다른 자판기가 꺼져있을 때 재고 복구 성공 확인
<input type="checkbox"/>	32	결함	완료	낮음	CPT-48 : 재고가 없고 다른 자판기가 꺼져있을 때 재고 복구 성공 확인
<input type="checkbox"/>	31	결함	완료	낮음	CPT-47 : 재고가 있고 다른 자판기가 꺼져있을 때 재고 복구 성공 확인
<input type="checkbox"/>	30	결함	완료	낮음	CPT-44 : 선결제를 통해 구매 과정 중 다른 자판기에서 결제를 통해 재고가 0이 되었을 경우 환불이 실행되며 잔고와 복구가 되는 지 확인
<input type="checkbox"/>	29	결함	완료	낮음	CPT-33 : 다른 자판기에 재고가 있으나 꺼져있을 때 잔고와 재고가 정상적으로 차감되며 선결제 화면 전환 실패 확인
<input type="checkbox"/>	28	결함	완료	낮음	CPT-32 : 다른 자판기에 재고가 있으나 꺼져있을 때 잔고와 재고가 정상적으로 차감되며 선결제 실패 확인
<input type="checkbox"/>	27	결함	완료	낮음	CPT-31 : 다른 자판기에 재고가 있고 꺼져있을 때 인증코드가 중복없이 생성되며 잔고와 재고가 정상적으로 차감되는 선결제 성공하는 것 확인
<input type="checkbox"/>	26	결함	완료	낮음	CPT-30 : 다른 자판기에 재고가 있고 꺼져있을 때 잔고와 재고가 정상적으로 차감되는 선결제 성공 확인
<input type="checkbox"/>	25	결함	완료	낮음	BFT_30 : 환불 수신 기능
<input type="checkbox"/>	19	결함	완료	보통	CPT-43 : 재고가 없고 다른 자판기가 꺼져있으며 재고가 없을 때 인증코드 입력을 통한 구매 성공 확인
<input type="checkbox"/>	18	결함	완료	보통	CPT-42 : 재고가 없고 다른 자판기가 꺼져있으며 재고가 없을 때 인증코드 입력을 통한 구매 성공 확인
<input type="checkbox"/>	17	결함	완료	보통	CPT-41 : 재고가 없고 다른 자판기가 꺼져있으며 재고가 있을 때 인증코드 입력을 통한 구매 성공 확인
<input type="checkbox"/>	16	결함	완료	보통	CPT-40 : 재고가 없고 다른 자판기가 꺼져있으며 재고가 없을 때 인증코드 입력을 통한 구매 성공 확인
<input type="checkbox"/>	15	결함	완료	보통	CPT-39 : 재고가 없고 다른 자판기가 꺼져있으며 재고가 있을 때 인증코드 입력을 통한 구매 성공 확인
<input type="checkbox"/>	14	결함	완료	보통	CPT-38 : 재고가 없고 다른 자판기가 꺼져있으며 재고가 있을 때 인증코드 입력을 통한 구매 성공 확인
<input type="checkbox"/>	13	결함	완료	보통	CPT-37 : 재고가 있고 다른 자판기가 꺼져있을 때 인증코드 입력을 통한 구매 성공 확인
<input type="checkbox"/>	12	결함	완료	보통	CPT-36 : 재고가 있고 다른 자판기가 꺼져있을 때 인증코드 입력을 통한 구매 성공 확인
<input type="checkbox"/>	11	결함	완료	보통	CPT-24 : 연결된 다른 DVM이 존재하지 않은 상황에 재고가 없는 음료 선택시 화면 전환
<input type="checkbox"/>	10	결함	완료	보통	CPT-20 : 환불 요청시 Request Success and Msg type 8 전송 확인
<input type="checkbox"/>	9	결함	완료	보통	CPT-11 : 유저 모드 전환
<input type="checkbox"/>	8	결함	완료	보통	BFT_20 : 위치 설정
<input type="checkbox"/>	7	결함	완료	보통	BFT_24 : 자판기 종료 기능
<input type="checkbox"/>	1	결함	완료	보통	Spec Review 1차

## -일감에 등록된 결함 중 해결된 결함에 대한 완료 처리 ( CPT/BFT/ Spec Review)

```

developer Developer의 댓글:
developer Developer의 댓글:
admin Admin의 댓글:
| 관리자 모드와 유저 모드의 구분이 모호하여 테스트 불가능
관리자 모드는 스크린샷의 화면만 관리자 모드로 생각하고 spec을 작성하였습니다. spec 문서 어느 부분에서 모호하다고 판단하셨는지 알려주시면 수정에 큰 도움이 될 것 같습니다.
SRS에 추가로 명시하였습니다.
SRS 16p 3.2.6 Admin Mode 에 관리자 모드와 사용자 모드에 대해 구체적으로 명시되어 수정된 것을 확인하였습니다.
developer Developer의 댓글:
admin Admin의 댓글:
| 재고가 없는 음료 선택시 DVM List 로 단순히 화면전환, DVM이 아니면 사용중인 경우 재고가 없는 음료를 누를 때, 주문 불가능 안내 메시지 추가 필요 판단 - 개발팀과 협의
오류 문구 추가하였습니다.
오류 문구 작동 확인하였습니다.
admin Admin이(가) 2일 전에 변경
• 상태음(음) 영문에서 음리(으)로 변경되었습니다.

```

## ※ Redmine에 등록된 일부 일감의 내용 사진

-위와 같이 결함에 대한 개발팀의 댓글 기반으로 수정된 DVM 테스트 재 진행 후 결과 기록

### 1.6 Summary

- Spec Review

: 스펙 리뷰 기반으로 수정 확인 완료

- Category Partitioning Test

: Test Case : 54 -> 54/54 Passed ( Fail : 0 ) : 100 %

- Brute Force Test

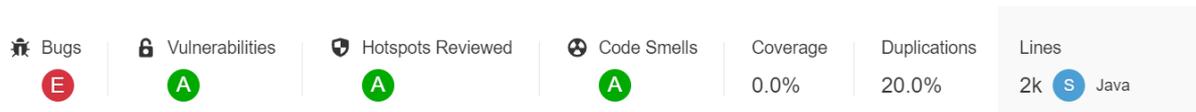
: Test Case : 35 -> 35/35 Passed ( Fail : 0 ) : 100 %

=2차 BFT / CPT 테스트 결과 A2팀의 DVM는 Spec 기반 잘 구현되었다고 판단 가능하다.

## 2 Static Analysis

### 2.1 SonarQube [ Bugs, Security Hotspots, CodeSmells]

#### [Overview]



※ Coverage는 Jacoco를 사용하여 2.4 Code Coverage 에서 작성

sonarqube ☆ master

Overview Issues Security Hotspots Measures Code Activity

Project Settings Project Information

My Issues All

Bulk Change

1 / 235 issues 3d 2h effort

Filters

Clear All Filters

Type CODE SMELL

Clear

Bug 10

Vulnerability 0

Code Smell 235

Severity

Blocker 0 Minor 133

Critical 39 Info 0

Major 63

Scope

Resolution

Status

Security Category

Creation Date NEW CODE

Clear

Language

Rule

src/main/java/GUI/GUI\_Admin.java

2 duplicated blocks of code must be removed. Why is this an issue? 15 days ago

Rename this package name to match the regular expression '^[a-z\_]+\.[a-z\_][a-z0-9]\*\$'. Why is this an issue? 15 days ago

Remove this unused import 'java.awt.EventQueue'. Why is this an issue? 15 days ago

Remove this unused import 'java.awt.event.ActionListener'. Why is this an issue? 15 days ago

Remove this unused import 'java.awt.event.ActionEvent'. Why is this an issue? 15 days ago

Rename this class name to match the regular expression '^[A-Z][a-zA-Z0-9]\*\$'. Why is this an issue? 15 days ago

Rename this field "gui\_frame" to match the regular expression '^[a-z][a-zA-Z0-9]\*\$'. Why is this an issue? 15 days ago

Rename this local variable to match the regular expression '^[a-z][a-zA-Z0-9]\*\$'. Why is this an issue? 15 days ago

※ 개발팀에게 알려야 할 필요가 있는 정적분석 결과만 기록하여 정리 및 Redmine 일감에 등록  
 - Bugs / Security Hotspots / CodeSmells [Critical | Major | Minor] 구분하여 정리 및 기록

[Bug] – 10개

File	Line	Rule	Review
Network.java	96	Bug-1 : Null pointers should not be dereferenced	Line 25 에서 Msg sendMsg = null 이 라고 선언하고 Line 96에서 sendMsg.getSrc_id()를 하는 것은 NullPointerException 에러가 발생할 가능성이 있다. 따라서 NullPointerException 처리가 필요하다.
	31	Bug-6 : Resources should be closed	위 리소스는 Closeable 해야 합니다. 닫지 않는다면 예외로 인해 호출이 수행되지 않을 수 있습니다. AutoCloseable 클래스 'try-with-resources'패턴을 사용하는 것을 추천합니다.
	112	Bug-7 : Resources should be closed	위 리소스는 Closeable 해야 합니다. 닫지 않는다면 예외로 인해 호출이 수행되지 않을 수 있습니다. AutoCloseable 클래스 'try-with-resources'패턴을 사용하는 것을 추천합니다.
	141	Bug-8 : Resources should be closed	위 리소스는 Closeable 해야 합니다. 닫지 않는다면 예외로 인해 호출이 수행되지 않을 수 있습니다. AutoCloseable 클래스 'try-with-resources'패턴을 사용하는 것을 추천합니다.
	178	Bug-9 : Resources should be closed	위 리소스는 Closeable 해야 합니다. 닫지 않는다면 예외로 인해 호출이 수행되지 않을 수 있습니다. AutoCloseable 클래스 'try-with-resources'패턴을 사용하는 것을 추천합니다.
	208	Bug-10 : Resources should be closed	위 리소스는 Closeable 해야 합니다. 닫지 않는다면 예외로 인해 호출이 수행되지 않을 수 있습니다. AutoCloseable 클래스 'try-with-resources'패턴을 사용하는 것 또는 'Soccket'을 'finally'안에 넣는 것을 추천합니다.

Controllor.java	210	Bug-5 : Resources should be closed	위 리소스는 Closeable 해야 합니다. 닫지 않는다면 예외로 인해 호출이 수행되지 않을 수 있습니다. AutoCloseable 클래스 'try-with-resources'패턴을 사용하는 것을 추천합니다.
	89	Bug-4 : Resources should be closed	위 리소스는 Closeable 해야 합니다. 닫지 않는다면 예외로 인해 호출이 수행되지 않을 수 있습니다. AutoCloseable 클래스 'try-with-resources'패턴을 사용하는 것을 추천합니다.
	38	Bug-3 : Resources should be closed	위 리소스는 Closeable 해야 합니다. 닫지 않는다면 예외로 인해 호출이 수행되지 않을 수 있습니다. AutoCloseable 클래스 'try-with-resources'패턴을 사용하는 것을 추천합니다.
CardList.java	28	Bug-2 : Resources should be closed	위 리소스는 Closeable 해야 합니다. 닫지 않는다면 예외로 인해 호출이 수행되지 않을 수 있습니다. AutoCloseable 클래스 'try-with-resources'패턴을 사용하는 것을 추천합니다.

**[Hotspots] -10개**

File	Line	Rule	Review	
Network.java	102	Security Hotspot-7 : printStackTrace()	printStackTrace()는 공격자의 공격을 용이하게 합니다.	
	151	Security Hotspot-8 : printStackTrace()		
	196	Security Hotspot-9 : printStackTrace()		
	224	Security Hotspot-10 : printStackTrace()		
Controller.java	71	Security Hotspot-3 : printStackTrace()		
	84	Security Hotspot-4 : printStackTrace()		
	100	Security Hotspot-6 : printStackTrace()		
CardList.java	52	Security Hotspot-2 : printStackTrace()		
VerificationCodeList.java	37	Security Hotspot-1 : Math.random()		Math.randome() java.lang.Math.random() method는 보안이 취약합니다. java.security.SecureRandom RNG 사용을 추천합니다.

[CodeSmells] - [Critical | Major | Minor]

File	Line	Rule	Severity	Review
Network.java	40, 96, 112	Code Smell	Critical	Define a constant instead of duplicating this literal
	49	Code Smell	Critical	Add a default case to this switch.
	L239-284 , L127	Code Smell	Major	코드 가독성을 높이기 위해 사용하지 않은 함수는 주석처리 대신 삭제하는 것이 좋다.
	L21	Code Smell	Major	상속 받아오는 class인 Function에서와 동일한 이름의 함수 run()을 선언했으므로 @Override를 추가해주어야 한다.
Controller.java	L252	Code Smell	Major	코드 가독성을 높이기 위해 사용하지 않은 함수는 주석처리 대신 삭제하는 것이 좋다.
GUI_Frame.java	139, 174	Code Smell	Critical	Add a default case to this switch.
	L45	Code Smell	Major	코드 가독성을 높이기 위해 사용하지 않은 함수는 주석처리 대신 삭제하는 것이 좋다.
GUI_Main.java	92	Code Smell	Critical	Use static access with "javax.swing.SwingConstants" for "CENTER".
GUI_VMList.java	L33	Code Smell	Major	코드 가독성을 높이기 위해 사용하지 않은 함수는 주석처리 대신 삭제하는 것이 좋다.
	L59	Code Smell	Minor	!(address[i]==null) 표현 보다는 address[i] !=null 표현으로 더 단순하게 표현하는 것이 좋다.
Card.java	L27-40	Code Smell	Major	코드 가독성을 높이기 위해 사용하지 않은 함수는 주석처리 대신 삭제하는 것이 좋다.

VM.java	L11	Code Smell	Major	코드 가독성을 높이기 위해 사용하지 않은 함수는 주석처리 대신 삭제하는 것이 좋다.
VMList.java	L33, L42, L51-54, L58, L80	Code Smell	Major	코드 가독성을 높이기 위해 사용하지 않은 함수는 주석처리 대신 삭제하는 것이 좋다.
GUI_Error.java	65	Code Smell	Critical	Add a default case to this switch.
	L13	Code Smell	Minor	선언시 초기화된 상수는 final보다 static final로 선언해야한다.
GUI_Dispenser.java	L14	Code Smell	Minor	선언시 초기화된 상수는 final보다 static final로 선언해야한다.
GUI_Admin.java	40	Code Smell	Critical	Define a constant instead of duplicating this literal
GUI_CardPayment.java	46	Code Smell	Critical	Define a constant instead of duplicating this literal
	38	Code Smell	Critical	Refactor this method to reduce its Cognitive Complexity
	62, 90, 118, ...	Code Smell	Critical	Add a nested comment explaining why this method is empty, throw an UnsupportedOperationException or complete the implementation.
GUI_VerificationCodeMode.java	28	Code Smell	Critical	Refactor this method to reduce its Cognitive Complexity
	54, 83, ...	Code Smell	Critical	Add a nested comment explaining why this method is empty, throw an UnsupportedOperationException or complete the implementation.
GUI_Prepay.java	36	Code Smell	Critical	Refactor this method to reduce its Cognitive Complexity
	60, 80, ...	Code Smell	Critical	Add a nested comment explaining why this method is empty, throw an UnsupportedOperationException or

				complete the implementation.
--	--	--	--	------------------------------

## 2.2 Redmine 일감

※ 개발팀에게 알려야 할 필요가 있는 정적분석 결과만 Redmine 일감에 등록

- 각 Bugs / Hotspots / CodeSmells 카테고리를 나누어서 Redmine에 등록

### [Bugs]

#### 일감

검색조건

상태      진행중 ▾  
 Testing      이다 ▾      Bugs ▾

옵션

적용     지우기     저장

<input type="checkbox"/>	# ▾	유형	상태	우선순위	제목
<input type="checkbox"/>	59	결함	신규	긴급	Bug-10 : Resources should be closed
<input type="checkbox"/>	57	결함	신규	긴급	Bug-9 : Resources should be closed
<input type="checkbox"/>	56	결함	신규	긴급	Bug-8 : Resources should be closed
<input type="checkbox"/>	55	결함	신규	긴급	Bug-7 : Resources should be closed
<input type="checkbox"/>	54	결함	신규	긴급	Bug-6 : Resources should be closed
<input type="checkbox"/>	53	결함	신규	긴급	Bug-5 : Resources should be closed
<input type="checkbox"/>	52	결함	신규	긴급	Bug-4 : Resources should be closed
<input type="checkbox"/>	51	결함	신규	긴급	Bug-3 : Resources should be closed
<input type="checkbox"/>	50	결함	신규	긴급	Bug-2 : Resources should be closed
<input type="checkbox"/>	49	결함	신규	보통	Bug-1 : Null pointers should not be dereferenced

### [Security Hotspots]

## 일감

▼ 검색조건

상태  진행중 ▼

Testing  이다 ▼  Security Hotspots ▼

→ 옵션

✓ 적용 🔄 지우기 📄 저장

<input type="checkbox"/>	#	유형	상태	우선순위	제목
<input type="checkbox"/>	48	결함	신규	낮음	Security Hotspot-10 : printStackTrace()
<input type="checkbox"/>	47	결함	신규	낮음	Security Hotspot-9 : printStackTrace()
<input type="checkbox"/>	46	결함	신규	낮음	Security Hotspot-8 : printStackTrace()
<input type="checkbox"/>	45	결함	신규	낮음	Security Hotspot-7 : printStackTrace()
<input type="checkbox"/>	44	결함	신규	낮음	Security Hotspot-6 : printStackTrace()
<input type="checkbox"/>	43	결함	신규	낮음	Security Hotspot-5 : printStackTrace()
<input type="checkbox"/>	42	결함	신규	낮음	Security Hotspot-4 : printStackTrace()
<input type="checkbox"/>	41	결함	신규	낮음	Security Hotspot-3 : printStackTrace()
<input type="checkbox"/>	40	결함	신규	낮음	Security Hotspot-2 : printStackTrace()
<input type="checkbox"/>	39	결함	신규	보통	Security Hotspot-1 : Math.random()

## [Code Smells]

### 일감

▼ 검색조건

상태  진행중 ▼  검색조건 추가

Testing  이다 ▼  Code Smells ▼

→ 옵션

✓ 적용 🔄 지우기 📄 저장

<input type="checkbox"/>	#	유형	상태	우선순위	제목
<input type="checkbox"/>	74	결함	신규	보통	[Critical] Use static access with "javax.swing.SwingConstants" for "CENTER".
<input type="checkbox"/>	73	결함	신규	보통	[Critical] Add a default case to this switch.
<input type="checkbox"/>	72	결함	신규	보통	[Critical] Add a nested comment explaining why this method is empty, throw an UnsupportedOperationException or complete the implementation.
<input type="checkbox"/>	70	결함	신규	보통	[Critical] Refactor this method to reduce its Cognitive Complexity
<input type="checkbox"/>	69	결함	신규	보통	[Critical] Define a constant instead of duplicating this literal "받은 고딕", "Receive Msg:", "Send Msg:", "localhost"
<input type="checkbox"/>	68	결함	신규	낮음	[Minor] Use the opposite operator ("!=") instead. - GUI_VMList.java
<input type="checkbox"/>	67	결함	신규	낮음	[Minor] Make this final field static too - GUI_Dispenser.java
<input type="checkbox"/>	66	결함	신규	낮음	[Minor] Make this final field static too - GUI_Error.java
<input type="checkbox"/>	65	결함	신규	낮음	[Major] This block of commented-out lines of code should be removed. - VMList.java
<input type="checkbox"/>	64	결함	신규	낮음	[Major] This block of commented-out lines of code should be removed. - VM.java
<input type="checkbox"/>	63	결함	신규	낮음	[Major] This block of commented-out lines of code should be removed. - Card.java
<input type="checkbox"/>	62	결함	신규	낮음	[Major] This block of commented-out lines of code should be removed. - GUI_VMList.java
<input type="checkbox"/>	61	결함	신규	낮음	[Major] This block of commented-out lines of code should be removed. - GUI_Frame.java
<input type="checkbox"/>	60	결함	신규	낮음	[Major] This block of commented-out lines of code should be removed. - Controller.java
<input type="checkbox"/>	58	결함	신규	낮음	[Major] This block of commented-out lines of code should be removed. - Network.java

### 결함 #68

#### [Minor] Use the opposite operator ("!=") instead. - GUI\_VMList.java

admin Admin이(가) 4일 전에 추가함.

상태:	신규	시작시간:
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.

### 결함 #42

#### Security Hotspot-4 : printStackTrace()

admin Admin이(가) 4일 전에 추가함.

상태:	신규
우선순위:	낮음
담당자:	-
Testing:	Security Hotspots

#### 설명

위치 : Controller.java / Line 84

printStackTrace()는 공격자의 공격을 용이하게 합니다.

#### Bug-5 : Resources should be closed

admin Admin이(가) 4일 전에 추가함.

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

#### 설명

위치 : Controller.java / Line : 210

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.
    }
}
```

-위 그림과 같이 Redmine 일감 내에 정적분석 오류위치와 설명 추가

## 2.3 Summary



## 2.4 Code Coverage - Jacoco

### [Overview]

#### Overall Coverage Summary

Package	Class, %	Method, %	Line, %
all classes	21.7% (15/ 69)	26.6% (62/ 233)	18.8% (253/ 1344)

#### Coverage Breakdown

Package ^	Class, %	Method, %	Line, %
GUI	0% (0/ 50)	0% (0/ 132)	0% (0/ 774)
Logic	80% (8/ 10)	52.3% (34/ 65)	29.7% (113/ 381)
tests	77.8% (7/ 9)	77.8% (28/ 36)	74.1% (140/ 189)

### [Logic]

#### Coverage Summary for Package: Logic

Package	Class, %	Method, %	Line, %
Logic	80% (8/ 10)	52.3% (34/ 65)	29.7% (113/ 381)

Class ^	Class, %	Method, %	Line, %
Beverage	100% (1/ 1)	100% (3/ 3)	100% (7/ 7)
Card	100% (1/ 1)	100% (5/ 5)	94.4% (17/ 18)
CardList	100% (1/ 1)	66.7% (2/ 3)	74.3% (26/ 35)
Controller	0% (0/ 1)	0% (0/ 19)	0% (0/ 117)
Msg	100% (1/ 1)	100% (5/ 5)	100% (10/ 10)
Network	0% (0/ 1)	0% (0/ 7)	0% (0/ 130)
VM	100% (1/ 1)	100% (6/ 6)	100% (13/ 13)
VMList	100% (1/ 1)	77.8% (7/ 9)	69.6% (16/ 23)
VerificationCode	100% (1/ 1)	100% (3/ 3)	100% (6/ 6)
VerificationCodeList	100% (1/ 1)	60% (3/ 5)	81.8% (18/ 22)

- Logic 내의 Controller.java, Network.java 파일 커버리지 계산을 위한 유닛 테스트 파일인 ControllerTest.java 의 테스트 실행시 제대로 테스트가 불가능하며 커버리지 계산과정에서 멈추는 문제 발생
- 해당 유닛 테스트가 불가능하다고 판단, 해당 파일을 제외한 나머지 부분에 대해 진행
- 그 외에 전체적인 클래스 파일에 대해 Unit Test Case의 커버리지가 높음을 볼 수 있으며 잘 짜여진 Unit Test Code 파일이라고 판단

## [Test]

Coverage Summary for Package: tests

Package	Class, %	Method, %	Line, %
tests	77.8% (7/ 9)	77.8% (28/ 36)	74.1% (140/ 189)

Class ^	Class, %	Method, %	Line, %
BeverageTest	100% (1/ 1)	100% (3/ 3)	100% (8/ 8)
CardListTest	100% (1/ 1)	100% (2/ 2)	100% (14/ 14)
CardTest	100% (1/ 1)	100% (5/ 5)	100% (23/ 23)
ControllerTest	0% (0/ 1)	0% (0/ 6)	0% (0/ 46)
MsgTest	100% (1/ 1)	100% (6/ 6)	100% (24/ 24)
VMListTest	100% (1/ 1)	100% (4/ 4)	100% (30/ 30)
VMTTest	100% (1/ 1)	100% (5/ 5)	100% (18/ 18)
VerificationCodeListTest	100% (1/ 1)	100% (3/ 3)	100% (23/ 23)
VerificationCodeTest	0% (0/ 1)	0% (0/ 2)	0% (0/ 3)

- ControllerTest.java 파일을 제외하고 진행하여 커버리지가 0이 나옴을 확인
- VerificationCodeTest.java 파일의 경우 내부 @Test 붙은 함수가 존재하지 않으며 객체만 생성하기 때문에 커버리지가 0이 나옴을 확인