

System Test & Spec Review

for Distributed Vending Machine(DVM) Class B - T2

T6

201511273 여찬종

201513205 송승현

201612368 이지우

201811301 한지희

CONTENTS

- 01 Refine CTIP
- 02 Specification Review
- 03 Structural Coverage – Jacoco
- 04 Category-Partition Testing
- 05 Brute Force Testing

01 Refine CTIP

Refine CTIP

System Testing Tool



Project: teamvv_testing
SonarQube id: teamvv_testing (2021)

Quality Gate: **FAILED**

Metric	Value
Size	XS (23 lines of code)
Reliability	A (0 bugs)
Security	A (0 vulnerabilities)
Maintainability	C (11 code smells, 1h 21min technical debt, 11.7% technical debt ratio)
Duplications	0%
Coverage	0%

konjuk_VV_t_6_2021
소프트웨어 프로젝트

프로젝트 / konjuk_VV_t_6_2021 / KVT62 보드

칸반 보드

Issue ID	Status	Cycle
KVT62-30	Spec Review	1st Cycle
KVT62-33	Brute Force	1st Cycle
KVT62-11	Strings and Boxed types should be compared using "equals()"	1st Cycle
KVT62-20	Source files should not have any duplicated blocks	1st Cycle
KVT62-34	BF1. 11 DVMS	1st Cycle
KVT62-12	Math operands should be cast before assignment	1st Cycle
KVT62-40	Functional-Requirements add 1	1st Cycle
KVT62-35	BF2. Show 20 and Sell 7 item	1st Cycle
KVT62-40	Functional-Requirements add 2	1st Cycle
KVT62-35	BF5. Left one item access - Cash	1st Cycle

02 Specification Review

Spec Review 진행한 Document

[OOAD 개발팀 Class B - T2]

Stage	Document Name
	DVM_PFR
1000	OOPT Stage1000 Project Planning Ver.2
SRS	SRS Ver.2
2030	OOPT Stage2030 OOA Ver.3
SDS	SDS Ver.2
2040	OOPT Stage2040 OOD Ver.2

Specification Review List - 1

No	Stage	Page	Name	Review
1	1000	3	Functional-Requirements add 1	DVM_PFR Page 2의 1.2 Requirements의 1번째 항목에서 '총 음료의 개수는 20종류이다.'에 대한 언급이 없으므로 추가 필요
2	1000	3	Functional-Requirements add 2	DVM_PFR Page 2의 1.2 Requirements의 2번째 항목에서 '한 자판기는 7종류의 음료를 판매한다.'에 대한 언급이 없으므로 추가 필요
3	1000	3	Functional-Requirements Refine 1	5번째 항목에서 다른 위치의 자판기 정보 제공하는 조건에 다른 자판기에는 해당 상품에 대한 재고가 있다는 조건 추가 필요
4	1000	3	Non-Functional Requirements add 1	DVM_PFR Page 3의 1.5 Assumptions and dependencies의 1번째 항목에서 '자판기의 개수는 최대 10개로 가정한다.'에 대한 언급이 없으므로 추가 필요
5	1000	3	Non-Functional-Requirements Refine 1	'자판기의 조작이 쉬워야한다', '재고의 관리가 용이해야 한다.', '자판기가 튼튼해야 한다.'와 같은 모호한 표현 수정 필요
6	1000	5	Functional Requirement - Description Refine	Pay by Cash, Pay by Card의 '가격합'이라는 표현과 Calculate Price(Cash), Calculate Price(Card)의 '상품들의'라는 표현을 보면 다중 선택이 가능하다는 의미가 될 수 있으므로 수정 필요
7	1000	5	Functional Requirement - Login	cf. switch가 무슨 뜻인지 이해할 수 없으므로 삭제 또는 추가 설명 필요
8	1000	6	R1.1 name	Page 5의 Function 1번째 항목은 Show Item인데, Show Menu로 나와있으므로 수정 필요

Specification Review List - 2

No	Stage	Page	Name	Review
9	1000	8	R1.5 Category	Page 6의 R1.5 Category는 Evident인데, Hidden으로 나와있으므로 수정 필요
10	1000	11	Use Case 15 Description	Page 14의 Test Case 15에는 '거리순서대로'라는 조건에 맞는 경우 Test Pass라고 나와있는데, 해당 조건에 대한 설명이 없으므로 추가 필요
11	SRS		Page Number	Page Number 추가 필요
12	2030	13	Term Name - card	Stage1000의 Page 7의 Credit Description과 동일하지만, Card라고 나와있으므로 수정 필요
13	2030	3	Use Case 1 - Overview 1	세 가지 주요 기능이 무엇인지 정확히 명시하도록 수정 필요
	SRS	5		
14	2030	3	Use Case 1 - Overview 2	Stage1000의 Page 9의 Description에는 '시스템이 상시 DVM에 있는 모든 제품의 이름과 재고 현황을 알려줍니다.'라고 나와있지만, '~ 초기 선택 메뉴를 보여줍니다.'로 불일치하는 설명 수정 필요
	SRS	5		
15	2030	3	Use Case 2	같은 의미의 용어로 보여지는 '제품', '상품'이 혼용되어 사용되고 있으므로 통일하는 방향으로 수정 필요
	SRS	5		
16	2030	9	Use Case 13 Type	Typical Courses of Events에 System의 Event만 존재하므로 Hidden으로 수정 필요
	SRS	11		

Specification Review List - 3

No	Stage	Page	Name	Review
17	2030 SRS	9 11	Use Case 14 Type	Typical Courses of Events에 System의 Event(Another DVM)만 존재하므로 Hidden으로 수정 필요
18	2030 SRS	11 12	Use Case Number	Stage1000의 Use Case는 19개로 17. Manage Kind of Product가 누락된 것이라면 수정 필요 (누락이 아닌 삭제의 경우, Stage1000 수정 필요)
19	2030	13	Refine Glossary	인증 코드라는 말을 주로 사용하지만 Glossary에는 코드에 대해서만 정의되어 있으므로 수정 또는 통일 필요
20	2030	16	System Sequence Diagram	Use Case [12-14]로 하나의 Sequence Diagram만 표현되어있는데, Use Case별로 분리하여 작성 필요
21	2030	21	Traceability System Operation	9. GetNoneltemLocation(), 10. GiveDVMInfo()에 대해서만 뒤에 '()'표시를 붙여 함수형로 표현하였는데, 다른 항목에 대해서도 동일하게 함수형으로 처리하거나 '()'표시 삭제 필요
22	SRS	14	Other Requirement	1번째 항목의 '자판기 내부의' 다음에 이어지는 문장이 없으므로 삭제 또는 수정 필요
23	2030	18-20	System Test Case	거의 모든 항목에서 '정상적으로', '제대로'라는 표현을 사용하는데 어떤 상황이 정상적인지 파악 불가하므로 명확한 용어 사용 필요
24	2040	8	Use Case 10 - Cross Reference	Stage2030 Use Case 10의 내용과 불일치하므로 삭제 또는 수정 필요

Specification Review List - 4

No	Stage	Page	Name	Review
25	2040	9	Use Case 11 - Cross Reference	Stage2030 Use Case 11의 내용과 불일치하므로 삭제 또는 수정 필요
26	2040 SDS	15-17 6-8	Interaction Diagram	Use Case별로 Interaction Diagram 분리 작성 필요
27	2040	16	Use Case 16 - Event	Typical Courses of Events에 1번 Event 뒤에 아이디와 비밀번호 출력에 대한 System의 Event가 누락되어 있으므로 추가 필요
28	2040 SDS	16 7	Interaction Diagram - Use Code & Code Validation	CodeTable의 경우 Class Diagram에서는 객체가 아닌데, Interaction Diagram에서는 객체인 것처럼 작성되어 있음
29	2040	21	Traceability line	연결하는 화살표가 많이 겹쳐서 내용 파악이 불가하므로 넘버링을 통해 표로 작성 요청
30	SDS		Page Number	Page Number 추가 필요
31	SDS	6	Interaction Diagram - 3.3.3	Calculate Price에 '(Cash)'가 누락되어 추가 필요

Spec Review No.1 / No.2 / No.4

DVM_PFR

1.2 Requirements

- 총 음료의 개수는 20 종류이다.
- 한 자판기는 7 종류의 음료를 판매한다.

1.5 Assumptions and dependencies

- 각 자판기는 모두 네트워크에 연결되어 있고 네트워크 연결 정보는 미리 알고 있다고 가정한다.
 - 자판기의 개수는 최대 10개로 가정한다.



Stage1000

전혀 고려하지 않고 있어 수정 요구

Spec Review No.14

Stage1000

Use Case	1. Show Menu
Actors	System
Description	- 시스템이 상시 DVM에 있는 모든 제품의 이름과 재고 현황을 알려줍니다

SRS

Stage2030

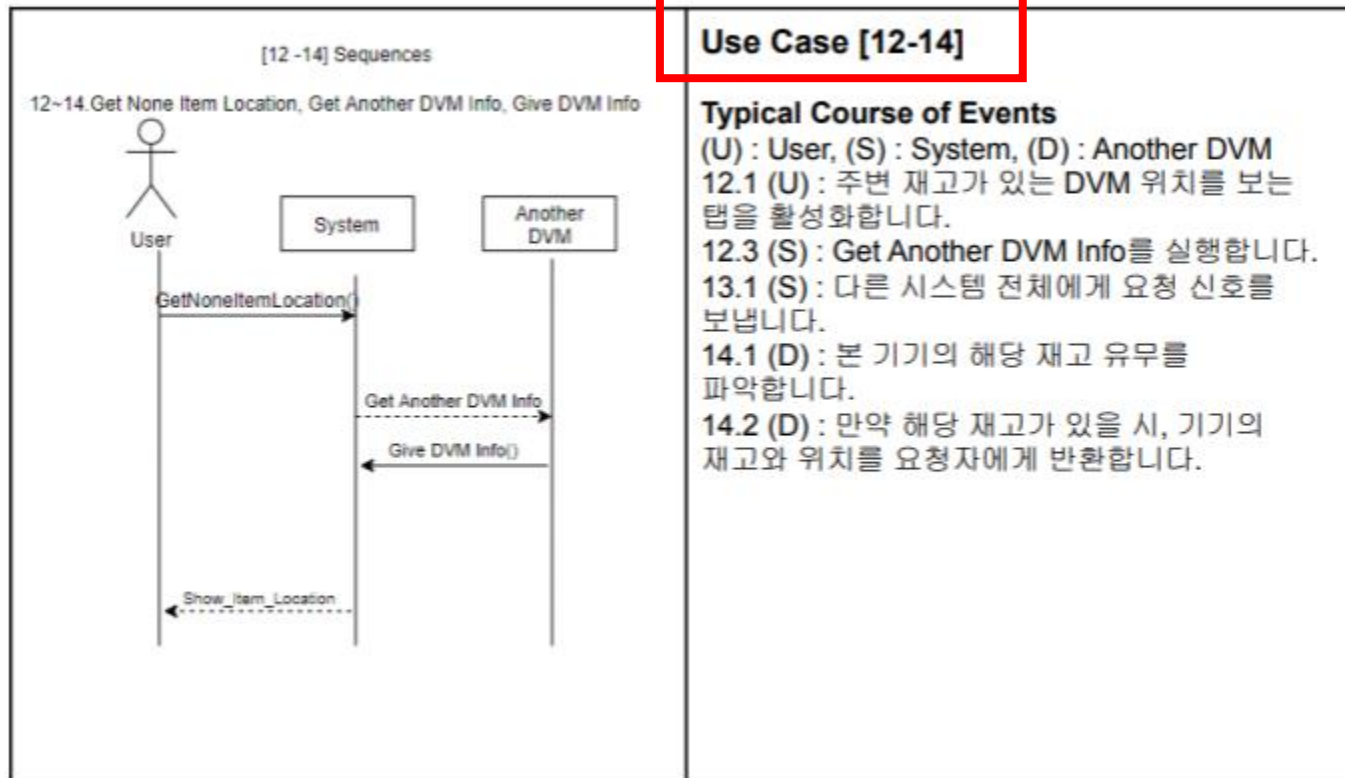
Use Case	1. Show Menu
Actor	System
Purpose	사용자에게 초기 선택 화면을 보여줍니다.
Overview	사용자에게 세가지 주요 기능 중 하나를 선택할 수 있는 초기 선택 메뉴를 보여줍니다.



불일치하는 설명 수정 필요

Spec Review No.20

Stage2030

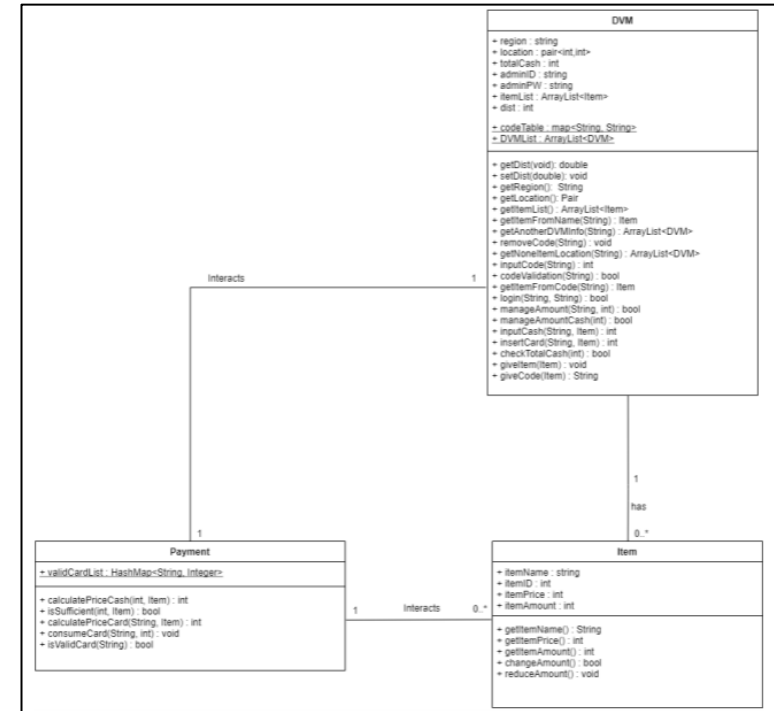
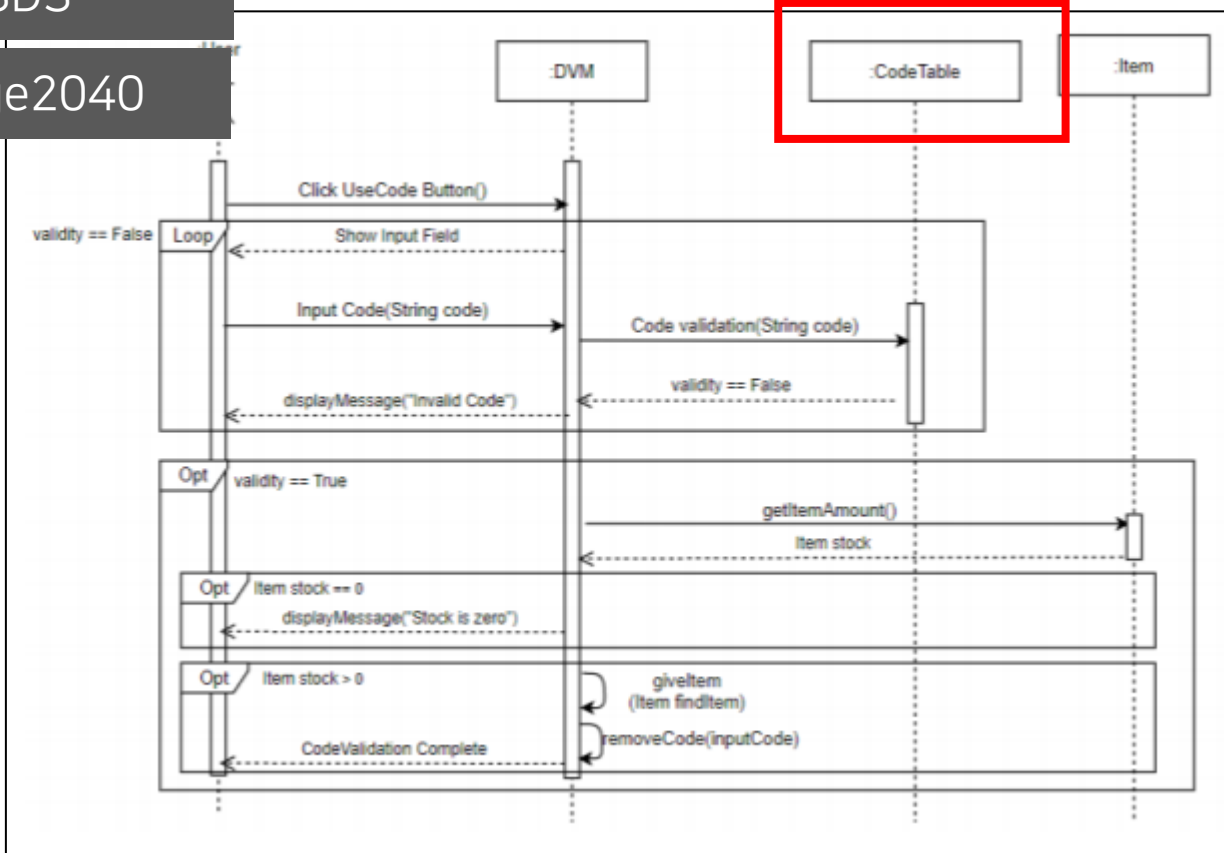


Use Case 12-14의
Type이 모두 Evident로
Sequence Diagram
Use Case별로 분리하여 작성 필요

Spec Review No.28

SDS

Stage2040



➔ CodeTable의 경우 Class Diagram에서는 객체가 아닌데, Interaction Diagram에서는 객체인 것처럼 작성되어 있어 수정 요구

CTIP 환경 - Jira에 추가

프로젝트 / konkuk_VV_t_6_2021 / KVT62-29 / KVT62-30

Spec Review

첨부 하위 작업 생성 이슈 연결

설명
설명 추가하기...

링크된 이슈

blocks

<input checked="" type="checkbox"/>	KVT62-40	Functional-Requirements add 1	↑	SPECREVIEW
<input checked="" type="checkbox"/>	KVT62-41	Functional-Requirements add 2	↑	SPECREVIEW
<input checked="" type="checkbox"/>	KVT62-43	Non-Functional Requirements add 1	↑	SPECREVIEW
<input checked="" type="checkbox"/>	KVT62-44	Non-Functional-Requirements Refine 1	↑	SPECREVIEW
<input checked="" type="checkbox"/>	KVT62-45	Functional Requirement - Description Refine	↑	SPECREVIEW
<input checked="" type="checkbox"/>	KVT62-46	Functional Requirement - Login	↑	SPECREVIEW
<input checked="" type="checkbox"/>	KVT62-48	R1.5 Category	↑	SPECREVIEW
<input checked="" type="checkbox"/>	KVT62-47	R1.1 name	↑	SPECREVIEW
<input checked="" type="checkbox"/>	KVT62-49	Use Case 15 Description	↑	SPECREVIEW
<input checked="" type="checkbox"/>	KVT62-50	Page Number	↑	SPECREVIEW
<input checked="" type="checkbox"/>	KVT62-51	Term Name - card	↑	SPECREVIEW
<input checked="" type="checkbox"/>	KVT62-53	Use Case 1 - Overview 2	↑	SPECREVIEW
<input checked="" type="checkbox"/>	KVT62-52	Use Case 1 - Overview 1	↑	SPECREVIEW
<input checked="" type="checkbox"/>	KVT62-42	Functional-Requirements Refine 1	↑	SPECREVIEW
<input checked="" type="checkbox"/>	KVT62-54	Use Case 2	↑	SPECREVIEW
<input checked="" type="checkbox"/>	KVT62-55	Use Case 13 Type	↑	SPECREVIEW
<input checked="" type="checkbox"/>	KVT62-58	Refine Glossary	↑	SPECREVIEW
<input checked="" type="checkbox"/>	KVT62-57	Use Case Number	↑	SPECREVIEW
<input checked="" type="checkbox"/>	KVT62-59	System Sequence Diagram	↑	SPECREVIEW
<input checked="" type="checkbox"/>	KVT62-56	Use Case 14 Type	↑	SPECREVIEW
<input checked="" type="checkbox"/>	KVT62-61	Other Requirement	↑	SPECREVIEW
<input checked="" type="checkbox"/>	KVT62-63	Use Case 10 - Cross Reference	↑	SPECREVIEW

프로젝트 / konkuk_VV_t_6_2021 / KVT62 보드

칸반 보드

내 이슈만 최근 업데이트됨

디버깅이슈 18 스펙리뷰 32 시스템테스팅 7 진행 중 1 완료 2

급행 4개 이슈

나머지 기타 56개 이슈

Spec Review 1st Cycle KVT62-30	Brute Force 1st Cycle KVT62-33	1st Cycle 1st Cycle KVT62-29	Category-Partition 1st Cycle KVT62-31
Strings and Boxed types should be compared using "equals()" KVT62-11	BF1. 11 DVMs 1st Cycle KVT62-34	CTIP 구축 CTIP 구축 KVT62-2	
Math operands should be cast before assignment KVT62-12	BF2. Show 20 and Sell 7 item 1st Cycle KVT62-35	최근에 수정된 이슈만 표시하고 있습니다. 오래된 이슈를 찾으십니까?	
Double Brace Initialization should not be used KVT62-13	BF5. Left one item access - Cash 1st Cycle KVT62-36		
Fields in a "Serializable" class should			

03 Structural Coverage - Jacoco

Jacoco Counter

Jacoco Counter

Instructions Counter	Jacoco의 최소단위로 나누어진 코드의 실행여부로 계산
Branches Counter	조건문으로 나누어지는 분기의 실행여부로 계산



Unit Test Coverage

Code를 각 Counter의 단위로 나누었을 때,

개발팀이 작성한 Unit Test Code가 실행한 Code의 비율

(Coverage 수식은 (실행된 각 Code단위 수) / (전체 Code단위 수) X 100로 계산함.)

Coverage - 전체 Code














































OOADProject				
Element	Missed Instructions	Cov.	Missed Branches	Cov.
default		19%		61%
Total	2,902 of 3,586	19%	48 of 124	61%

Instructions Coverage

$$684/3,586 \times 100 = 19\%$$

Branches Coverage

$$76/124 \times 100 = 61\%$$

default				
Element	Missed Instructions	Cov.	Missed Branches	Cov.
Main		0%		n/a
CardGUI.new ActionListener()		0%		0%
CashGUI.new ActionListener()		0%		0%
ItemListGUI		0%		0%
ManageAmountGUI		0%		n/a
NotEnoughAmountGUI		0%		n/a
ManageCashGUI		0%		n/a
MainGUI		0%		n/a
LoginGUI		0%		n/a
SelectPaymentGUI		0%		n/a
UseCouponGUI		0%		n/a
CardGUI		0%		n/a
CashGUI		0%		n/a
AdminMenuGUI		0%		n/a
NotEnoughAmountGUI.new ActionListener()		0%		0%
UseCouponGUI.new ActionListener()		0%		0%
ItemListGUI.new ActionListener()		0%		0%
ManageAmountGUI.new ActionListener()		0%		0%
ManageCashGUI.new ActionListener()		0%		0%
LoginGUI.new ActionListener()		0%		0%
DVM		90%		90%
SelectPaymentGUI.new ActionListener()		0%		n/a
SelectPaymentGUI.new ActionListener()		0%		n/a
NotEnoughAmountGUI.new ActionListener()		0%		n/a
ManageCashGUI.new ActionListener()		0%		n/a
LoginGUI.new ActionListener()		0%		n/a
CardGUI.new ActionListener()		0%		n/a
CashGUI.new ActionListener()		0%		n/a
ManageAmountGUI.new ActionListener()		0%		n/a
MainGUI.new ActionListener()		0%		n/a
MainGUI.new ActionListener()		0%		n/a
ItemListGUI.new ActionListener()		0%		n/a
UseCouponGUI.new ActionListener()		0%		n/a
NotEnoughAmountGUI.new ActionListener()		0%		n/a
AdminMenuGUI.new ActionListener()		0%		n/a
AdminMenuGUI.new ActionListener()		0%		n/a
MainGUI.new ActionListener()		0%		n/a
AdminMenuGUI.new ActionListener()		0%		n/a
SelectPaymentGUI.new ActionListener()		0%		n/a
ManageCashGUI.new KeyAdapter()		0%		0%
ManageAmountGUI.new KeyAdapter()		0%		0%
DVM.Pair		66%		n/a
Payment		100%		100%
Item		100%		100%
Payment.new HashMap()		100%		n/a
Total	2,902 of 3,586	19%	48 of 124	61%

Coverage – DVM, Payment, Item

DVM				
Element	Missed Instructions	Cov.	Missed Branches	Cov.
giveCode(Item)		55%		50%
getRegion()		0%		n/a
getLocation()		0%		n/a
DVM(String, int, int, int, String, String, ArrayList, double, Payment)		100%		87%
getNoneItemLocation(String)		100%		100%
getAnotherDVMInfo(String)		100%		100%
manageAmount(String, int)		100%		100%
getItemFromCode(String)		100%		100%
getItemFromName(String)		100%		100%
giveItem(Item)		100%		100%
inputCode(String)		100%		100%
login(String, String)		100%		100%
inputCash(int, Item)		100%		100%
removeCode(String)		100%		100%
manageAmountCash(int)		100%		100%
static {...}		100%		n/a
checkTotalCash(int)		100%		100%
insertCard(String, Item)		100%		n/a
sortDVM(ArrayList)		100%		n/a
codeValidation(String)		100%		n/a
setDist(double)		100%		n/a
getDist()		100%		n/a
getItemList()		100%		n/a
getCodeTable()		100%		n/a
Total	52 of 525	90%	6 of 66	90%









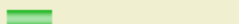


Instructions Coverage

$$684/3,586 \times 100 = 19\%$$

Branches Coverage

$$76/124 \times 100 = 61\%$$

Payment

Element	Missed Instructions	Cov.	Missed Branches	Cov.
calculatePriceCard(String, Item)		100%		100%
calculatePriceCash(int, Item)		100%		100%
consumeCard(String, int)		100%		n/a
isSufficient(int, int)		100%		100%
isValidCard(String)		100%		n/a
static (...)		100%		n/a
calculateChange(int, int)		100%		n/a
Payment()		100%		n/a
Total	0 of 90	100%	0 of 12	100%










Instructions Coverage

90/90 X 100 = 100%

Branches Coverage

12/12 X 100 = 100%

Item

Element	Missed Instructions	Cov.	Missed Branches	Cov.
Item()		100%		n/a
Item(String, int, int, int)		100%		n/a
reduceAmount()		100%		100%
changeAmount(int)		100%		100%
getItemName()		100%		n/a
getItemPrice()		100%		n/a
getItemAmount()		100%		n/a
Total	0 of 58	100%	0 of 4	100%

Instructions Coverage

58/58 X 100 = 100%

Branches Coverage

4/4 X 100 = 100%

04 Category-Partition Testing

Categorize

Phase No.	Phase
1	Activate_Choices
2	Admin_Choices
3	Admin_Login
4	Change_Cash
5	Choose_Item
6	Manage_Amount
7	Not_In_Stock_select
8	Pay By Card
9	Pay by Cash
10	Pay By Code
11	Payment Choices

Phase 1-7,11은 화면 구성에 따라 구분

Phase 8-10은 결제 방식에 따라 구분

Phase	Group	Category
Phase Name	Environment	Program Start
	Input	Button 또는 키보드를 통한 입력
	Value	Output

각 Phase는 위와 같은 Group과 Category로 구분
(일부 Phase에서는 입력을 Check할 수 있는 Group존재)

Category-Partition Categorize - 1

No	Phase	Group	Category	Value	Constraint
1	Activate_Choices	Environment	Program Start	Success	[property SC]
		Input	Button	Buy Product	[if SC]
				Use Code	[if SC]
				Admin Login	[if SC]
		Value	Output	True	
				False	
2	Admin_Choices	Environment	Program Start	Success	[property SC]
		Input	Button	Manage Stock	[if SC] [property MS]
				Manage Cash	[if SC] [property MC]
				Cancel	[if SC] [property C]
		Value	Output	Move to Manage Stock	[if MS]
				Move to Manage Cash	[if MC]
				Move to Back	[if C]

Category-Partition Categorize - 2

No	Phase	Group	Category	Value	Constraint
3	Admin_Login	Environment	Program Start	Success	[property SC]
		Input	Button1	ID,PW Input	[if SC] [property IDPW]
				Cancel	[if SC] [property C]
			Button2	Enter	[if SC] [if !C] [property Enter]
			Admin Info	valid	[if IDPW] [property valid]
		invalid		[if IDPW] [property invalid]	
		Value	Output	Move to admi Selection	[if valid]
				ID or Password incorrect	[if invalid]
				Move to Back	[if C]
4	Change_Cash	Environment	Program Start	Success	[property SC]
		Input	Button1	Cash Input	[if SC] [property IDPW]
				Cancel	[if SC] [property C]
			Button2	Enter	[if SC] [if !C] [property Enter]
			Number	Input is Integer	[if SNI] [property valid]
		Input is not Integer		[if SNI] [property invalid]	
		Value	Output	Successful Change	[if SNI] [if valid]
				input is not Integer	[if SNI && invalid]
				Move to Back	[if C]
			Cash	Same as Input	[if SC] [if SNI && valid] [if !C && !invalid]

Category-Partition Categorize - 3

No	Phase	Group	Category	Value	Constraint
5	Choose_Item	Environment	Program Start	Success	[property SC]
		Input	Item	In Stock Item	[if SC] [property ISI]
				Not In Stock Item	[if SC] [property NISI]
				Cancel	[if SC] [property C]
		Value	Output	Move to Payment	[if ISI]
				Move to NISI select	[if NISI]
				Move to Back	[if C]
6	Manage Amount	Environment	Program Start	Success	[property SC]
		Input	Button1	Stock Name Input	[if SC] [property SNI]
				Cancel	[if SC] [property C]
			Button2	Stock Amount Input	[if SC] [property SAI]
			Button3	Enter	[if SC] [if !C] [property Enter]
			Name	Correct Name	[if SC] [if SNI] [if !C] [property validName]
				Incorrect Name	[if SC] [if SNI] [if !C] [property invalidName]
			Amount	Same as Input	[if SC] [if SNI] [if !C] [if validName]
		Value	Output	Successful Change	[if validName]
				Incorrect Name Input	[if invalidName]
				Move to Back	[if C]

Category-Partition Categorize - 4

No	Phase	Group	Category	Value	Constraint
7	Not_In_Stock_select	Environment	Program Start	Success	[property SC]
		Input	Button	Search Another DVM	[if SC] [property SADVM]
				Buy Code	[if SC] [property BC]
				Admin Login	[if SC] [property C]
		Value	Output	Pop up other DVM info	[if SADVM]
				Move to Payment	[if BC]
				Move to Back	[if C]
8	Pay By Card	Environment	Program Start	Success	[property SC]
		Input	Button1	Card Number Input	[if SC] [property CNI]
				Cancel	[if SC] [property C]
			Button2	Stock Amount Input	[if SC] [property SAI]
			Button3	Enter	[if SC] [if !C] [property Enter]
			Card Number Format	True	[if SC] [if CNI] [property ValidCard]
		False		[if SC] [if CNI] [property InValidCard]	
		Value	Output	Successful Buy	[if ValidCard]
				Fail to Buy	[if InValidCard]
				Move to Back	[if C]

Category-Partition Categorize - 5

No	Phase	Group	Category	Value	Constraint
9	Pay By Cash	Environment	Program Start	Success	[property SC]
		Input	Button1	Cash Input	[if SC] [property CI]
				Cancel	[if SC] [property C]
			Button2	Enter	[if SC] [if !C] [property Enter]
			Amount	Less Input	[if CI][property LeI]
				Same Input	[if CI][property SI]
				Larger Input	[if CI][property LI]
		Value	Output	Successful Buy	[if SI LI]
				Fail to Buy	[if LeI]
				Move to Back	[if C]
			Change	True	[if LI]
				False	[if !LI && !C]

Category-Partition Categorize - 6

No	Phase	Group	Category	Value	Constraint
10	Pay By Code	Environment	Program Start	Success	[property SC]
		Input	Button1	Code Number Input	[if SC] [property CNI]
				Cancel	[if SC] [property C]
			Button2	Enter	[if SC] [if !C] [property Enter]
			Code Number Format	True	[if SC] [if CNI] [property ValidCode]
		False		[if SC] [if CNI] [property InValidCode]	
		Value	Output	Successful Buy	[if ValidCode]
				Fail to Buy	[if InValidCode]
				Move to Back	[if C]
		11	Payment Choices	Environment	Program Start
Input	Button			Pay By Cash	[if SC] [property PBCash]
				Pay By Card	[if SC] [property PBCard]
				Cancel	[if SC] [property C]
Value	Output			Move to Pay by Cash	[if PBCash]
				Move to Pay by Card	[if PBCard]
				Move to Back	[if C]

Category-Partition Test Result

Phase No.	Phase	All Test	Constraint 적용	Passed/All	Coverage(%)
1	Activate_Choices	1x3x2 = 6	7	7/7	100
2	Admin_Choices	1x3x3 = 9	3	3/3	100
3	Admin_Login	1x2x1x2x3 = 12	2	2/2	100
4	Change_Cash	1x2x1x2x3x1 = 12	3	3/3	100
5	Choose_Item	1x3x3 = 9	3	3/3	100
6	Manage_Amount	1x2x1x1x2x1x3 = 12	3	3/3	100
7	Not_In_Stock_select	1x3x3 = 9	3	3/3	100
8	Pay By Card	1x2x1x1x2x3 = 12	3	3/3	100
9	Pay by Cash	1x2x1x3x3x2 = 36	4	4/4	100
10	Pay By Code	1x2x1x2x3 = 12	3	3/3	100
11	Payment Choices	1x3x3 = 9	3	3/3	100
		138	37	37/37	100

* 37개 중 37개 Pass (100%)

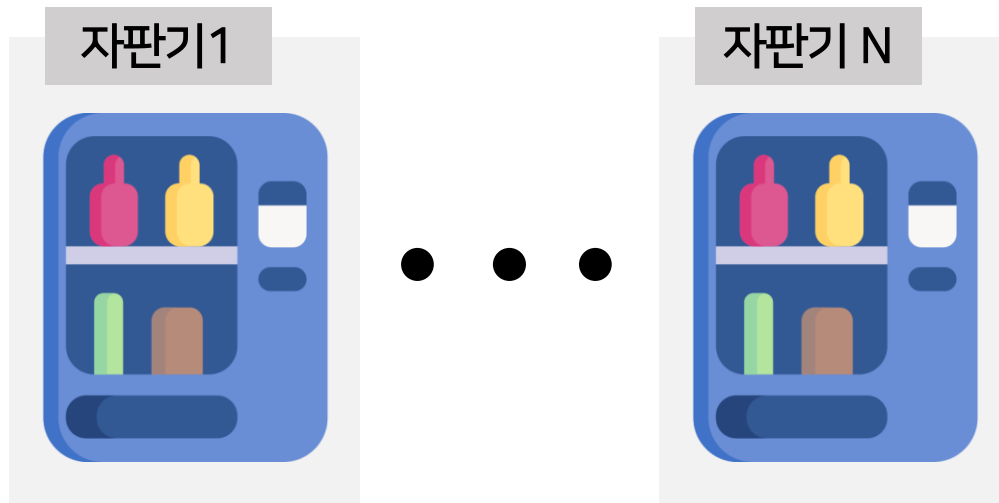
CTIP 환경 - Zephyr에 추가

KVT62-87	PASS	DVM-1.1	Category_Partition_Testing
KVT62-88	PASS	DVM-1.2	Category_Partition_Testing
KVT62-89	PASS	DVM-1.3	Category_Partition_Testing
KVT62-90	PASS	DVM-1.4	Category_Partition_Testing
KVT62-91	PASS	DVM-1.5	Category_Partition_Testing
KVT62-92	PASS	DVM-1.6	Category_Partition_Testing
KVT62-93	PASS	DVM-1.7	Category_Partition_Testing
KVT62-94	PASS	DVM-2.1	Category_Partition_Testing
KVT62-95	PASS	DVM-2.2	Category_Partition_Testing
KVT62-96	PASS	DVM-2.3	Category_Partition_Testing
KVT62-97	PASS	DVM-3.1	Category_Partition_Testing
KVT62-98	PASS	DVM-3.2	Category_Partition_Testing
KVT62-99	PASS	DVM-4.1	Category_Partition_Testing
KVT62-100	PASS	DVM-4.2	Category_Partition_Testing
KVT62-101	PASS	DVM-4.3	Category_Partition_Testing
KVT62-102	PASS	DVM-5.1	Category_Partition_Testing
KVT62-103	PASS	DVM-5.2	Category_Partition_Testing
KVT62-104	PASS	DVM-5.3	Category_Partition_Testing
KVT62-105	PASS	DVM-6.1	Category_Partition_Testing
KVT62-106	PASS	DVM-6.2	Category_Partition_Testing

KVT62-107	PASS	DVM-6.3	Category_Partition_Testing
KVT62-108	PASS	DVM-7.1	Category_Partition_Testing
KVT62-109	PASS	DVM-7.2	Category_Partition_Testing
KVT62-110	PASS	DVM-7.3	Category_Partition_Testing
KVT62-111	PASS	DVM-8.1	Category_Partition_Testing
KVT62-112	PASS	DVM-8.2	Category_Partition_Testing
KVT62-113	PASS	DVM-8.3	Category_Partition_Testing
KVT62-114	PASS	DVM-9.1	Category_Partition_Testing
KVT62-115	PASS	DVM-9.2	Category_Partition_Testing
KVT62-116	PASS	DVM-9.3	Category_Partition_Testing
KVT62-117	PASS	DVM-9.4	Category_Partition_Testing
KVT62-118	PASS	DVM-10.1	Category_Partition_Testing
KVT62-119	PASS	DVM-10.2	Category_Partition_Testing
KVT62-120	PASS	DVM-10.3	Category_Partition_Testing
KVT62-121	PASS	DVM-11.1	Category_Partition_Testing
KVT62-122	PASS	DVM-11.2	Category_Partition_Testing
KVT62-123	PASS	DVM-11.3	Category_Partition_Testing

05 Brute Force Testing

Brute Force Test Case 작성 시 고려사항



Category-Partition

개별 DVM Machine의 기능

Brute Force

DVM 사이의 관계와

기능 이외의 환경 조건에 대한 Test를 중심

Brute Force Test Case List - 1

No	Test Case	Description	Test Result
1	11 DVMs	자판기를 11개 만드는 게 불가능한지 확인한다.	Fail
2	Show 20 and Sell 7 item	자판기가 20개의 음료를 가지며 재고가 있는 음료가 7개 존재함을 확인한다.	Fail
3	Only One DVM Check non-item	자판기가 1개일 때, 해당 자판기의 음료A에 대한 재고가 없는 경우를 설정한다. 이 후 다른 자판기로의 재고 요청 시 주변에 구입가능한 자판기가 없다는 결과를 확인한다.	Pass
4	Two DVM check non-item	자판기가 2개일 때, 각 자판기의 음료A에 대한 재고가 없는 경우를 설정한다. 이 후 자판기 중 한 자판기에서 '다른 DVM확인'을 입력하는 경우 주위에 가능한 자판기가 없다는 결과를 확인한다.	Pass
5	Left one item access - Cash	음료A에 대한 재고가 없는 자판기1과 음료A에 대한 재고가 1개 남은 자판기2를 설정한다. 자판기1에서 음료A에 대한 구매를 누른 뒤, 자판기2에서 음료A를 현금으로 구매한다. 이 후 자판기 1에서 '다른 DVM확인'을 입력하는 경우 주위에 가능한 자판기가 없다는 결과를 확인한다.	Fail
6	Left one item access - Card	음료A에 대한 재고가 없는 자판기1과 음료A에 대한 재고가 1개 남은 자판기2를 설정한다. 자판기1에서 음료A에 대한 구매를 누른 뒤, 자판기2에서 음료A를 카드로 구매한다. 이 후 자판기 1에서 '다른 DVM확인'을 입력하는 경우 주위에 가능한 자판기가 없다는 결과를 확인한다.	Fail
7	Left one item access - Code	음료A에 대한 재고가 없는 자판기1과 음료A에 대한 재고가 1개 남은 자판기2를 설정한다. 자판기1에서 음료A에 대한 구매를 누른 뒤, 자판기2에서 음료A를 인증코드로 구매한다. 이 후 자판기 1에서 '다른 DVM확인'을 입력하는 경우 주위에 가능한 자판기가 없다는 결과를 확인한다.	Fail
8	Card limit - one DVM	한도가 4000인 카드로 자판기1에서 4000을 결제하고 1000인 음료A를 구매했을 때, 한도초과임을 확인한다.	Pass

Brute Force Test Case List - 2

No	Test Case	Description	Test Result
9	Card limit - Two DVM	한도가 4000인 카드로 자판기1에서 2000, 자판기 2에서 2000을 결제하고, 자판기3에서 결제 시 한도초과임을 확인한다.	Pass
10	Cash increase	관리자 메뉴에서 자판기의 현금보유량을 500으로 설정한 뒤 가격이 1000인 음료A를 구매했을 때, 현금보유량의 증가를 확인한다.	Fail
11	Code - One DVM	자판기1에서 재고가 없는 음료A에 대한 인증 코드를 구입한 뒤, 관리자 메뉴에서 자판기1의 음료A에 대한 재고를 추가한 뒤 인증 코드로 구매가 가능함을 확인한다.	Pass
12	Code - Two DVM	자판기1에서 재고가 없는 음료A에 대한 인증 코드를 구입한 뒤, 자판기2에서 음료A에 대해 인증 코드로 구매가 가능함을 확인한다.	Pass
13	One code two access	음료A에 대해 구매한 인증 코드를 자판기1과 자판기2에 동일하게 입력한 뒤 순차적으로 쿠폰번호 입력을 누르는 경우 자판기1에서만 구매가 가능함을 확인한다.	Pass
14	Item inventory check - Non-item	음료A에 대한 재고가 없는 자판기1과 음료A에 대한 재고가 1개 남은 자판기2를 설정한다. 자판기1에서 음료A에 대한 구매를 누른 뒤, 자판기2의 관리자 메뉴에서 음료A에 대한 재고를 0으로 수정한다. 이 후 자판기 1에서 '다른 DVM확인'을 입력하는 경우 주위에 가능한 자판기가 없다는 결과를 확인한다.	Fail
15	Item inventory check - Item	음료A에 대한 재고가 없는 자판기1, 자판기2와 음료A에 대한 재고가 2개 남은 자판기 3을 설정한다. 자판기1에서 음료A에 대한 구매를 누른 뒤, 자판기2의 관리자 메뉴에서 음료A에 대한 재고를 1으로 수정한다. 이 후 자판기 1에서 '다른 DVM확인'을 입력하는 경우 자판기2에 대한 거리 결과를 확인한다.	Fail

* 15개 중 7개 Pass (46.7%)

Brute Force Test Case No.1



Brute Force Test Case No.5



구매 가능한 자판기에
자판기2가 존재하므로
Test Fail

1. 음료A에 대한 구매 클릭

2. 음료A 구매

3. 구매 가능한 다른 DVM 확인

Brute Force Test Case No.10



1. 음료A 1000 구매 클릭
2. 현금으로 구매 클릭 후 구매
3. 관리자 메뉴 - 보유현금변경 클릭



보유 현금이 음료를 현금 구매 해도
달라지지 않으므로 Fail

CTIP 환경 - Jira에 추가

프로젝트 / konkuk_VV_t_6_2021 / KVT62-29 / KVT62-33

Brute Force

첨부 하위 작업 생성 이슈 연결

설명 • 저장되지 않은 변경 사항
System Testing 중 하나로 전체적인 System을 Test합니다.

링크된 이슈

is blocked by

- KVT62-34 BF1. 11 DVMs SYSTEMTESTING
- KVT62-35 BF2. Show 20 and Sell 7 item SYSTEMTESTING
- KVT62-36 BF5. Left one item access - Cash SYSTEMTESTING
- KVT62-37 BF6. Left one item access - Card SYSTEMTESTING
- KVT62-38 BF7. Left one item access - Code SYSTEMTESTING
- KVT62-39 BF10. Cash increase SYSTEMTESTING

Math operands should be cast before assignment KVT62-12

Double Brace Initialization should not be used KVT62-13

Fields in a "Serializable" class should

KVT62 보드

내 이슈만 최근 업데이트됨

스펙리뷰 32 시스템테스팅 7 진행 중 1 완료 2

Spec Review KVT62-30

Brute Force KVT62-33

1st Cycle KVT62-29

Category-Partition KVT62-34

BF1. 11 DVMs KVT62-34

BF2. Show 20 and Sell 7 item KVT62-35

BF5. Left one item access - Cash KVT62-36

CTIP 구축 KVT62-2

최근에 수정된 이슈만 표시하고 있습니다.
 오래된 이슈를 찾으십니까?

CTIP 환경 - Zephyr에 추가

*ID ^	*현황 v	*요약	*라벨(Label) v
KVT62-72	FAIL	11 DVMs	Brute_Force_Testing
KVT62-73	FAIL	Show 20 and Sell 7 item	Brute_Force_Testing
KVT62-74	PASS	Only One DVM Check non-item	Brute_Force_Testing
KVT62-75	PASS	Two DVM check non-item	Brute_Force_Testing
KVT62-76	FAIL	Left one item access - Cash	Brute_Force_Testing
KVT62-77	FAIL	Left one item access - Card	Brute_Force_Testing
KVT62-78	FAIL	Left one item access - Code	Brute_Force_Testing
KVT62-79	PASS	Card limit - one DVM	Brute_Force_Testing
KVT62-80	PASS	Card limit - Two DVM	Brute_Force_Testing
KVT62-81	FAIL	Cash increase	Brute_Force_Testing
KVT62-82	PASS	Code - One DVM	Brute_Force_Testing
KVT62-83	PASS	Code - Two DVM	Brute_Force_Testing
KVT62-84	PASS	One code two access	Brute_Force_Testing
KVT62-85	FAIL	Item inventory check - Non-item	Brute_Force_Testing
KVT62-86	FAIL	Item inventory check - Item	Brute_Force_Testing

Jira Board

프로젝트 / konkuk_VV_t_6_2021 / KVT62 보드

칸반 보드

릴리즈 | Test View | ...

🔍 👤 내 이슈만 최근 업데이트됨

디버깅이슈 69 스펙리뷰 33 시스템테스팅 7 진행 중 1 완료 2

▼ 급행 4개 이슈

	<p>Spec Review</p> <p>1st Cycle</p> <p>🟢 ↑ KVT62-30</p>	<p>Brute Force</p> <p>1st Cycle</p> <p>🟢 ↑ KVT62-33</p>	<p>1st Cycle</p> <p>1st Cycle</p> <p>🟢 ↑ KVT62-29</p>	<p>Category-Partition</p> <p>1st Cycle</p> <p>🟢 ↑ KVT62-34</p>
--	---	---	---	--

▼ 나머지 기타 108개 이슈

<p>Strings and Boxed types should be compared using "equals()"</p> <p>🟢 ↑ ... KVT62-11</p>	<p>Source files should not have any duplicated blocks</p> <p>🟢 ↑ KVT62-20</p>	<p>BF1. 11 DVMs</p> <p>1st Cycle</p> <p>🟢 ↑ KVT62-34</p>		<p>CTIP 구축</p> <p>CTIP 구축</p> <p>🟢 ↑ ... KVT62-2</p>
<p>Math operands should be cast before assignment</p> <p>🟢 ↑ ... KVT62-12</p>	<p>Functional-Requirements add 1</p> <p>1st Cycle</p> <p>🟢 ↑ KVT62-40</p>	<p>BF2. Show 20 and Sell 7 item</p> <p>1st Cycle</p> <p>🟢 ↑ KVT62-35</p>		<p>최근에 수정된 이슈만 표시하고 있습니다.</p> <p>🔍 오래된 이슈를 찾으시겠습니까?</p>
<p>Double Brace Initialization should not be used</p> <p>🟢 ↑ ... KVT62-13</p>	<p>Functional-Requirements add 2</p> <p>1st Cycle</p> <p>🟢 ↑ KVT62-41</p>	<p>BF5. Left one item access - Cash</p> <p>1st Cycle</p> <p>🟢 ↑ KVT62-36</p>		
<p>"static" base class members should</p>				

Jira Board - 1st Cycle Issue 진행률

프로젝트 / konuk_VV_t_6_2021 / KVT62-29

1st Cycle

첨부 에픽에 이슈 생성 이슈 연결

설명

- 5월 20일 검증팀에서 개발팀에게 작성한 문서에 대한 Spec Review와 System Testing 결과 제공
 - Issue 구분
 - Spec Review
 - System Testing
 - Category Partition
 - Brute Force
- 5월 28일 개발팀이 Issue 전체 해결하여 2nd Cycle 제출

이 에픽의 이슈 ... +

KVT62-30 Spec Review ↑ SPECREVIEW

KVT62-33 Brute Force ↑ SYSTEMTESTING

KVT62-31 Category-Partition ↑ 완료됨

2% 완료

KVT62-29 / KVT62-33

Brute Force

첨부 하위 작업 생성 이슈 연결

설명 • 저장되지 않은 변경 사항

System Testing 중 하나로 전체적인 System을 Test합니다.

링크된 이슈 +

is blocked by

<input checked="" type="checkbox"/>	KVT62-34 BF1. 11 DVMs	↑	SYSTEMTESTING
<input checked="" type="checkbox"/>	KVT62-35 BF2. Show 20 and Sell 7 item	↑	SYSTEMTESTING
<input checked="" type="checkbox"/>	KVT62-36 BF5. Left one item access - Cash	↑	SYSTEMTESTING
<input checked="" type="checkbox"/>	KVT62-37 BF6. Left one item access - Card	↑	SYSTEMTESTING
<input checked="" type="checkbox"/>	KVT62-38 BF7. Left one item access - Code	↑	SYSTEMTESTING
<input checked="" type="checkbox"/>	KVT62-39 BF10. Cash increase	↑	SYSTEMTESTING