

Distributed vending machine

학번	이름
201511238	허준호
201911147	강속영
201711320	오준엽
201511218	이지영
201710397	이지훈

Index

1. Class Diagram Change
2. JUnitTest
3. System Testing
4. Spec Review
5. Run Windows Change

1. Class Diagram Change

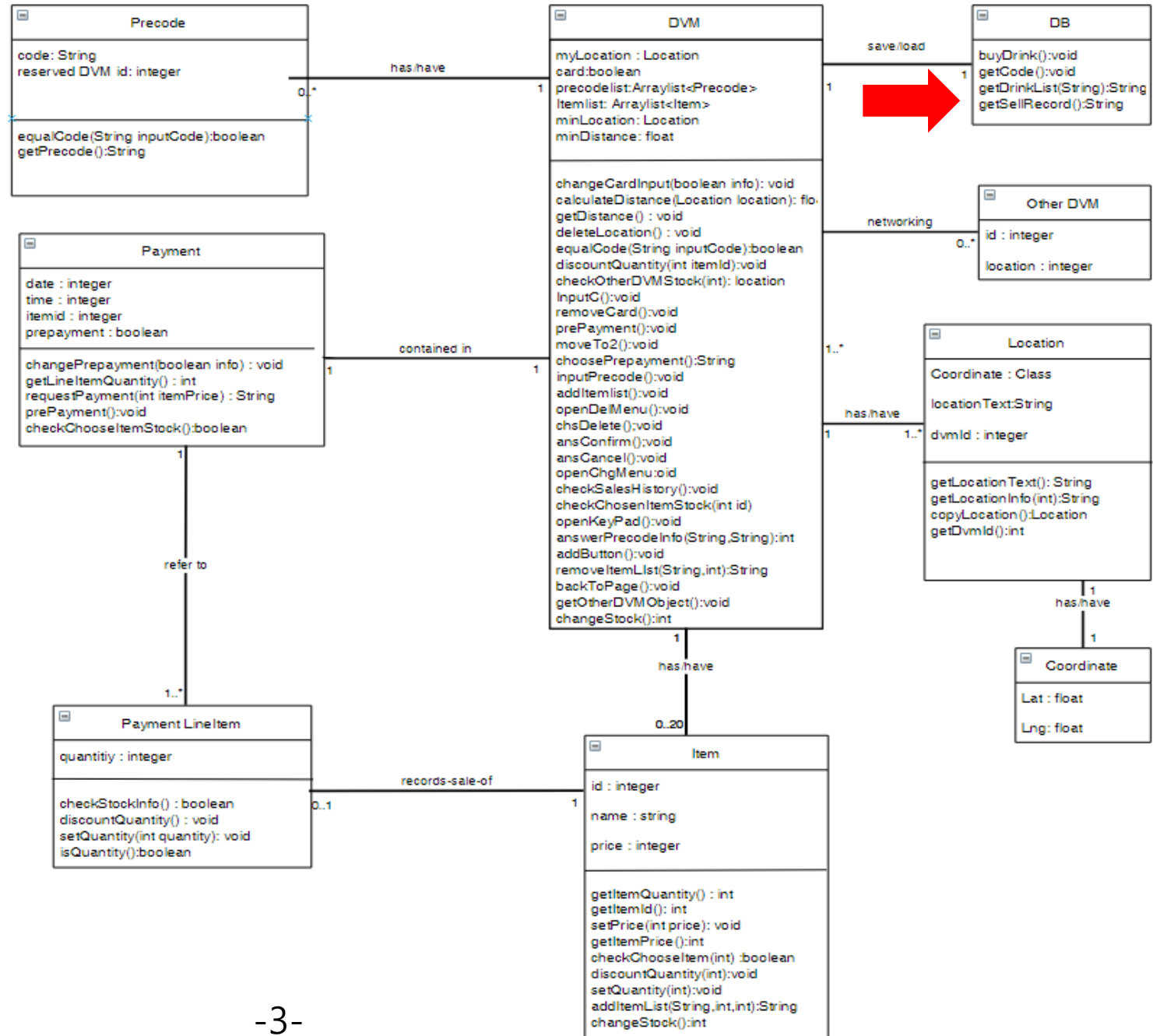
Class Diagram Change

JUnitTest

System Testing

Spec Review

Run Windows Change



1. Class Diagram Change

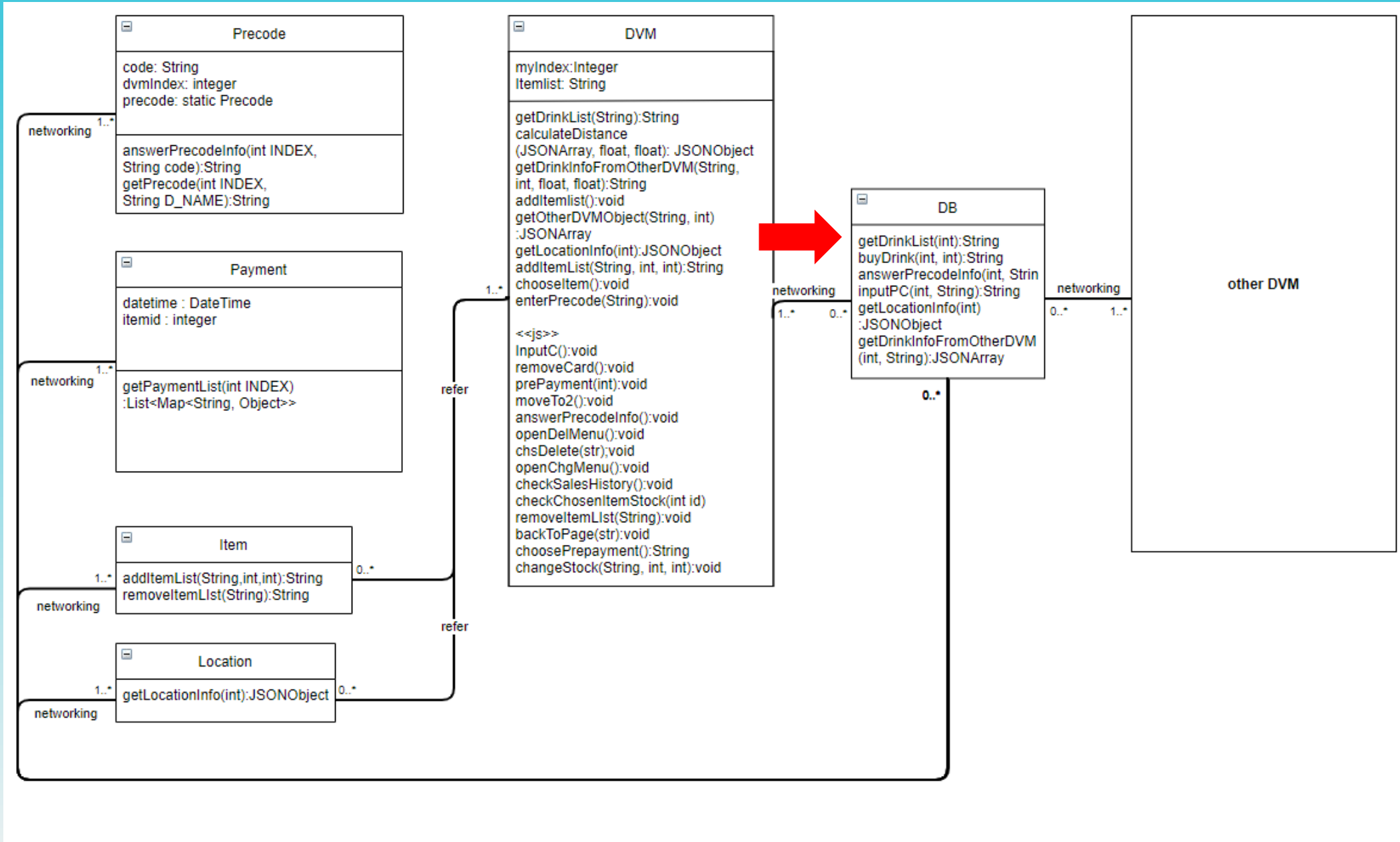
Class Diagram Change

JUnitTest

System Testing

Spec Review

Run Windows Change



2. JUnit Test

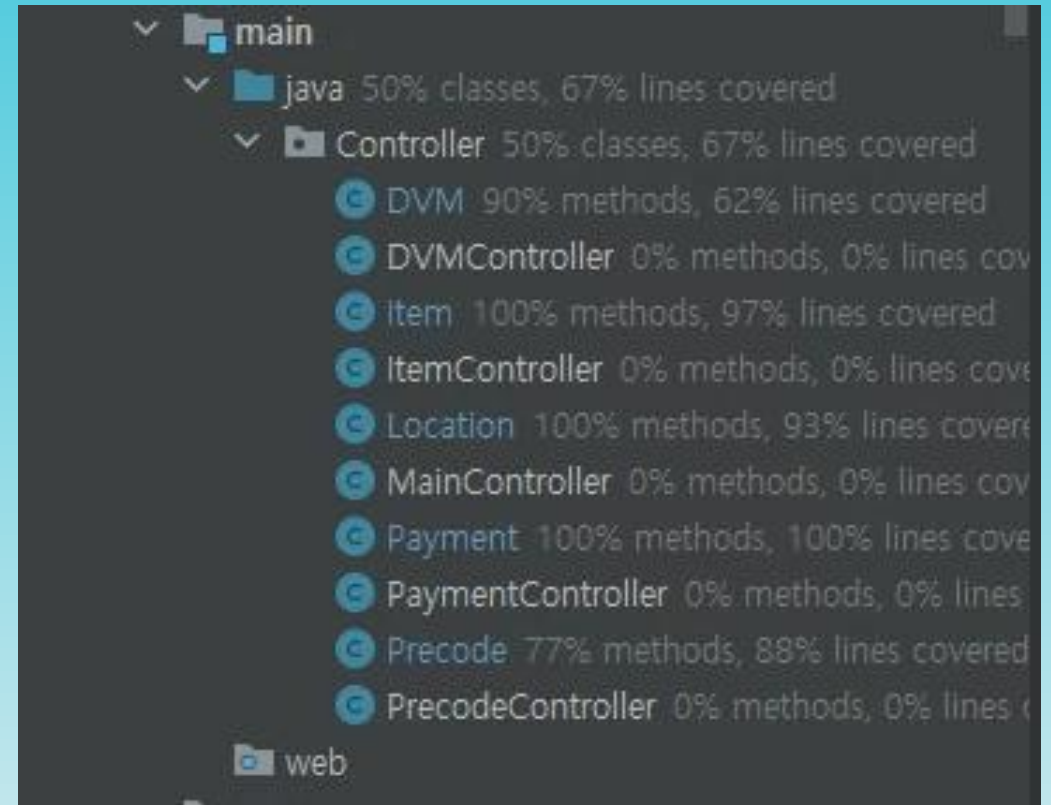
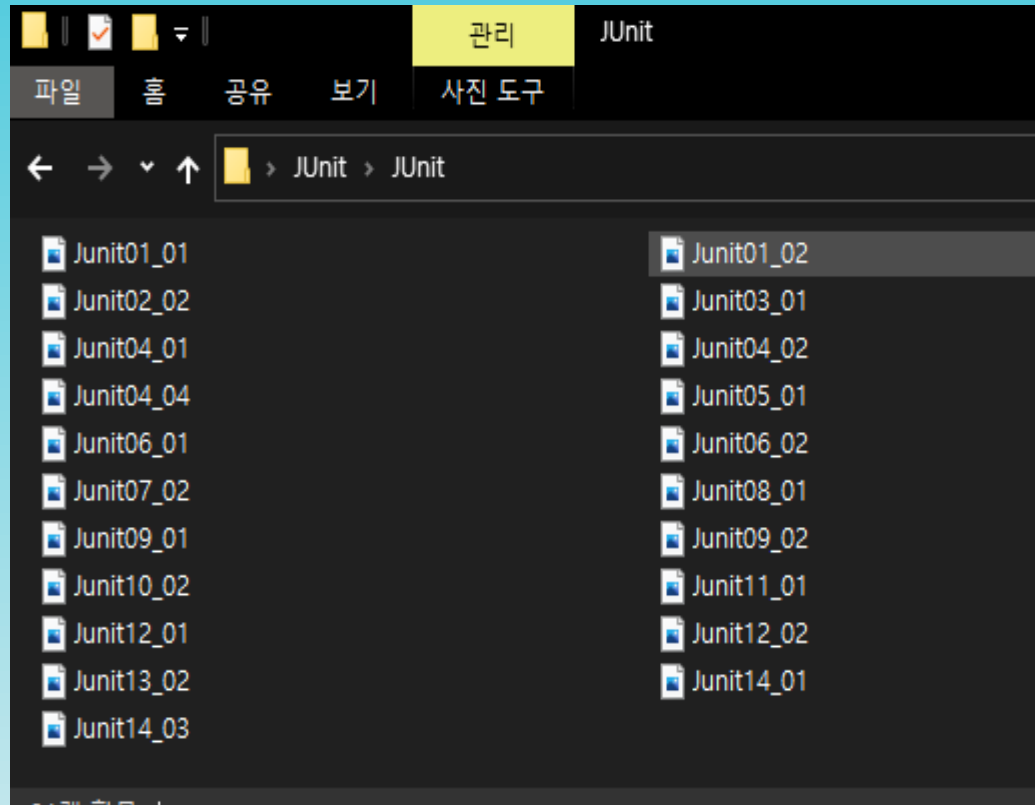
Class
Diagram
Change

JUnit
Test

System
Testing

Spec
Review

Run
Windows
Change



Coverage: SMA_Project [test] ×

50% classes, 67% lines covered in 'all classes in scope'

Element	Class, %	Method, %	Line, %
Controller	50% (5/10)	47% (22/46)	67% (262/390)

2. JUnit Test

```
28     int gpi = 1;
29     @RepeatedTest(10)
30     void getPrecodeTest() throws IOException, ParseException {
31         assertFalse(Precode.getPrecode(gpi++, D_NAME "").contains("HTTP Status 500 - Internal Server Error"));
32     }
```

✓ Test Results	128 ms
✓ PrecodeTest	128 ms
✓ getPrecodeTest()	128 ms
✓ repetition 1 of 10	35 ms
✓ repetition 2 of 10	13 ms
✓ repetition 3 of 10	13 ms
✓ repetition 4 of 10	10 ms
✓ repetition 5 of 10	10 ms
✓ repetition 6 of 10	9 ms
✓ repetition 7 of 10	8 ms
✓ repetition 8 of 10	12 ms
✓ repetition 9 of 10	10 ms
✓ repetition 10 of 10	8 ms

Class
Diagram
Change

JUnit
Test

System
Testing

Spec
Review

Run
Windows
Change

2. JUnit Test

```
14     int i = 0, price = 0;
15     @RepeatedTest(10)
16     void addItemlist() throws IOException {
17         assertFalse(Item.addItemlist( NAME: "", price, i++).contains("HTTP Status 500 - Internal Server Error"));
18         price += Math.abs(new Random().nextInt( bound: 10000));
19         System.out.println("price:"+price);
20     }
```

JUnit
Test

✓ Test Results	89 ms	"C:\Program Files\Java\jdk-11.0.11\bi
✓ ItemTest	89 ms	price:9677
✓ addItemlist()	89 ms	price:9339
✓ repetition 1 of 10	29 ms	price:9390
✓ repetition 2 of 10	7 ms	price:5056
✓ repetition 3 of 10	7 ms	price:4835
✓ repetition 4 of 10	6 ms	price:6099
✓ repetition 5 of 10	7 ms	price:9306
✓ repetition 6 of 10	6 ms	price:2026
✓ repetition 7 of 10	7 ms	price:5778
✓ repetition 8 of 10	7 ms	price:901
✓ repetition 9 of 10	6 ms	
✓ repetition 10 of 10	7 ms	
		Process finished with exit code 0

3. System Testing

The screenshot shows the GitHub interface for the repository **KUSVV-2021 / SMA_Project**. The navigation bar includes links for Pull requests, Issues, Marketplace, and Explore. The repository header shows 1 Watch, 0 Stars, and 0 Forks. The main navigation includes Code, Issues (173), Pull requests, Actions, Projects (2), Wiki, Security, and Insights. A search bar contains the text "is:open" and a "New project" button is visible. The Projects tab is active, displaying a list of 2 Open projects and 1 Closed project. The first project is **System Testing**, described as "문서의 내용과 결과물을 기반으로한 테스트 케이스들중 실패한 케이스들에 대한 이슈", updated 14 hours ago. The second project is **Spec Review**, described as "문서에 대한 이슈", also updated 14 hours ago. The footer contains copyright information for 2021 GitHub, Inc. and various links like Terms, Privacy, Security, Status, Docs, Contact GitHub, Pricing, API, Training, Blog, and About.

Class
Diagram
Change

JUnit
Test

System
Testing

Spec
Review

Run
Windows
Change

3. System Testing

KUSVV-2021 / SMA_Project

Watch 1

<> Code Issues 173 Pull requests Actions Projects 2 Wiki Security Insights Settings

System Testing

Updated 14 hours ago

Filter cards

0 To do

+ ...

26 In progress

+ ...

19 Wait for review

+ ...

0 Done

+ ...

[CP 1006] Not Enough Money
#148 opened by IMRaccoon
System Testing (1st cycle)

[CP 2004] Over Name Length
#150 opened by IMRaccoon
System Testing (1st cycle)

[CP 2005] Empty Length
#151 opened by IMRaccoon
System Testing (1st cycle)

[CP 2010] Input 0 in Price
#154 opened by IMRaccoon
System Testing (1st cycle)

[CP 2012] Over Range
#155 opened by IMRaccoon
System Testing (1st cycle)

[CP 2017] Over Range
#157 opened by IMRaccoon
System Testing (1st cycle)

[CP 2007] No Item
#152 opened by IMRaccoon
System Testing (1st cycle)

[CP 2008] Remove Item
#153 opened by IMRaccoon
System Testing (1st cycle)

[BF 0510] Input Precode 10
#175 opened by IMRaccoon
System Testing (1st cycle)

[BF 0501] Input Precode 01
#172 opened by IMRaccoon
System Testing (1st cycle)

[BF 0707] Add Item 07
#181 opened by IMRaccoon
System Testing (1st cycle)

[BF 0808] Remove Item 08
#189 opened by IMRaccoon
System Testing (1st cycle)

Automated as To do Manage

Automated as Done Manage

Class Diagram Change

JUnit Test

System Testing

Spec Review

Run Windows Change

3. System Testing

Important Change

1. 각종 수량에 대한 예외처리

[BF 0704] Add Item 04 #178

Open

IMRaccoon opened this issue 7 days ago · 1 comment



IMRaccoon commented 7 days ago

153 : [BF 0704] Add Item 04

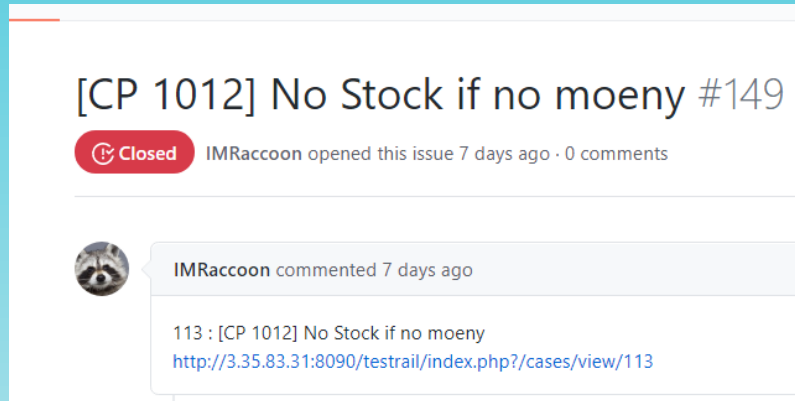
<http://3.35.83.31:8090/testrail/index.php?tests/view/153>

```
function addItemList() {  
    if ( ( document.getElementById( elementId: "name").value == "" ) //음료의 이름이 없을경우  
        || ( document.getElementById( elementId: "price").value == "" ) //가격이 비어있을 경우  
        || ( document.getElementById( elementId: "price").value == "0" ) //가격이 0이 입력된 경우  
        || ( document.getElementById( elementId: "price").value*1 > 50000 ) //가격이 5만원을 초과한 경우  
        || ( document.getElementById( elementId: "price").value*1 < 1 ) ) {  
        alert("잘못된 입력입니다.");  
        return;  
    } else {  
        if ( document.getElementById( elementId: "name").value.length > 20 ) {  
            alert("음료 이름의 길이는 20자를 넘길수 없습니다.");  
            return;  
        } else {  
            dvm.addItemList(document.getElementById( elementId: "name").value, document.getElementById( ele  
        }  
    }  
}
```

3. System Testing

Important Change

2. 문서에 기반한 구현 방법 수정 및 추가

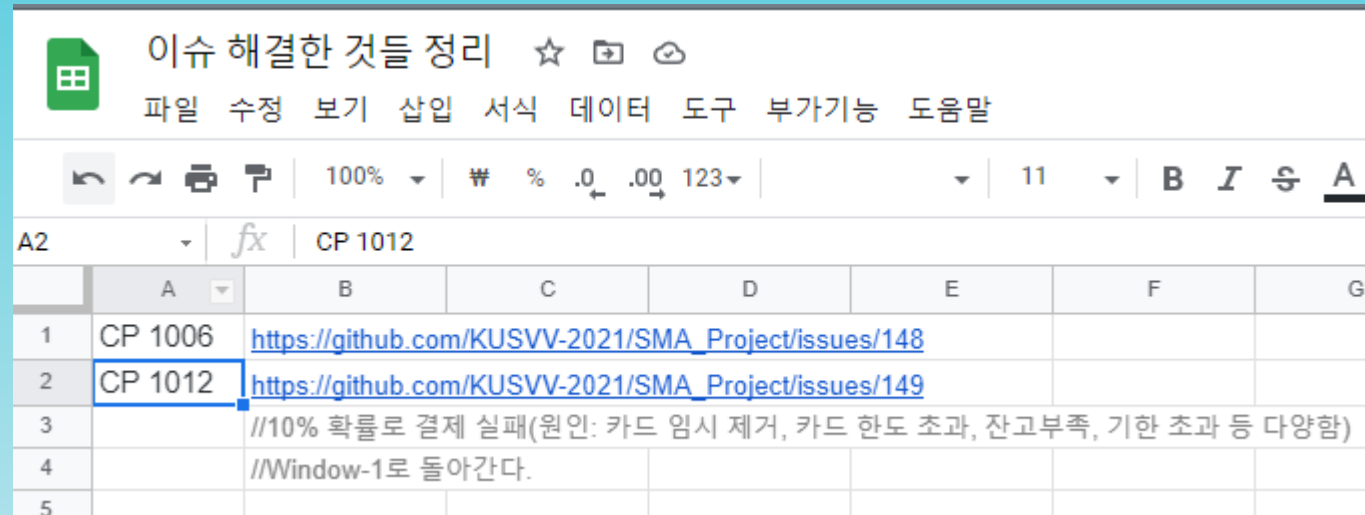


[CP 1012] No Stock if no moeny #149

Closed IMRaccoon opened this issue 7 days ago · 0 comments

IMRaccoon commented 7 days ago

113 : [CP 1012] No Stock if no moeny
<http://3.35.83.31:8090/testrail/index.php?cases/view/113>



이슈 해결한 것들 정리

파일 수정 보기 삽입 서식 데이터 도구 부가기능 도움말

100% # % .0 .00 123

A2 fx CP 1012

	A	B	C	D	E	F	G
1	CP 1006	https://github.com/KUSVV-2021/SMA_Project/issues/148					
2	CP 1012	https://github.com/KUSVV-2021/SMA_Project/issues/149					
3		//10% 확률로 결제 실패(원인: 카드 임시 제거, 카드 한도 초과, 잔고부족, 기한 초과 등 다양함)					
4		//Window-1로 돌아간다.					
5							

```
    alert("결제에 실패하였습니다.");  
}  
requestPayment(itemPrice) { //boolean  
    //일반적인 카드리더기에서는 잔고에 대한 내용을 알수없으므로 특정 확률에 따라 결제 실패와 성공여부를 판단  
    if (Math.floor(Math.random()*10) == 4) return false;  
    return true;  
}
```

Class
Diagram
Change

JUnit
Test

System
Testing

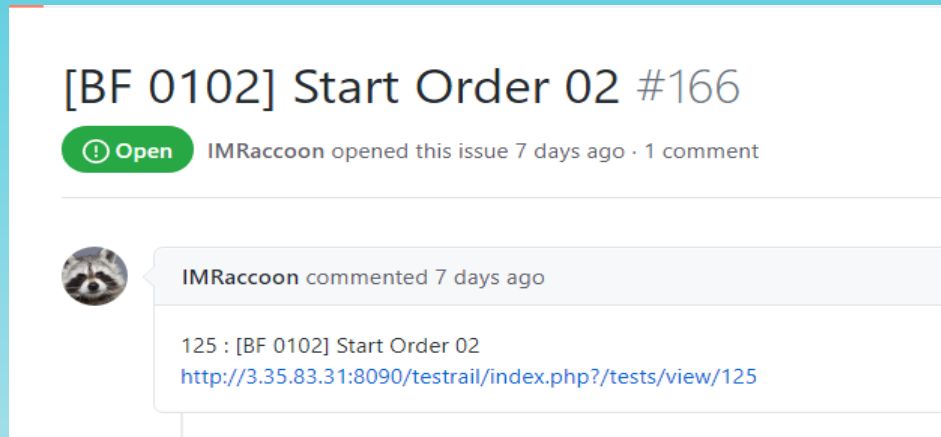
Spec
Review

Run
Windows
Change

3. System Testing

Important Change

2. 문서에 기반한 구현 방법 수정 및 추가



[BF 0102] Start Order 02 #166

Open IMRaccoon opened this issue 7 days ago · 1 comment

IMRaccoon commented 7 days ago

125 : [BF 0102] Start Order 02
<http://3.35.83.31:8090/testrail/index.php?/tests/view/125>

```
window.onkeydown = function (e : KeyboardEvent ) {  
    if (e.code == 'KeyR') {  
        dvm.removeCard();  
    }  
}
```

3. System Testing

Important Change

2. 구현 방법 수정 및 추가

	https://github.com/KUSVV-2021/SMA_Project/issues/177				
		버튼이 비활성화 된다는 문서 모두 수정부탁드립니다. 비활성화 x, 버튼클릭시 더이상 삭제할수없다는 문구			
	https://github.com/KUSVV-2021/SMA_Project/issues/178				
		같은 이름의 상품은 추가 불가능함, 재고만 추가됨			
BF 0705	https://github.com/KUSVV-2021/SMA_Project/issues/179				
		<code>alert("음료의 이름은 100자 이내로 작성가능합니다.");</code>			
BF 0802	https://github.com/KUSVV-2021/SMA_Project/issues/185				
BF 0803	https://github.com/KUSVV-2021/SMA_Project/issues/186				
BF 0807	https://github.com/KUSVV-2021/SMA_Project/issues/188				
		삭제기능 추가 (해당기기에서만 안보이게 state1이면 보이지않게 했음)			

Class
Diagram
Change

JUnit
Test

System
Testing

Spec
Review

Run
Windows
Change

4. Spec Review

1. 문서들간(Inception, OOA, OOD)의 통일

KUSVV-2021 / SMA_Project

Code Issues 173 Pull requests Actions Projects 2 Wiki Security Insights Settings

Spec Review
Updated 15 hours ago

Filter cards

To do (0)

In progress (1)

Wait for review (127)

Done (0)

[SR 3060] 2032 - 01 #100
Opened in KUSVV-2021/SMA_Project

yuz413 commented 8 days ago

Planning, Analysis, Design에서 각각 Use Case Name 통일시키고, 최종적으로 선택된 Use Case Name을 기반으로 Use Case Diagrams 수정

Automated as **To do** [Manage](#)

Automated as **Done** [Manage](#)

4. Spec Review

Important Change

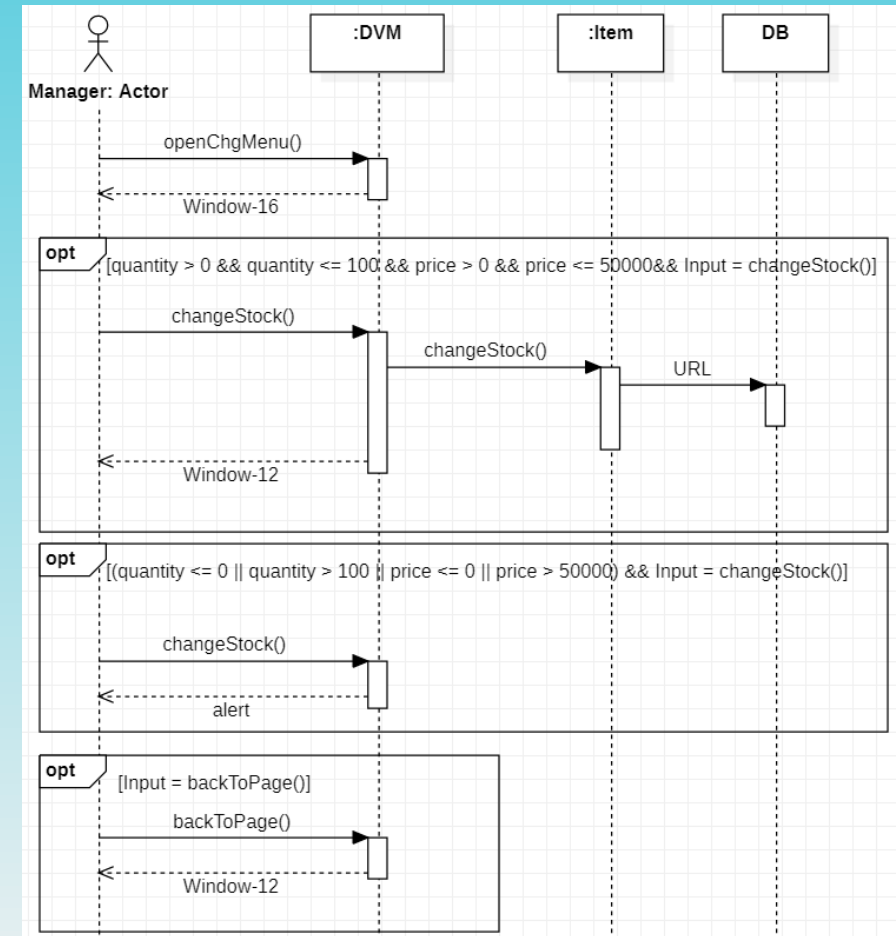
2. Use case의 통합

칠성사이다

현재 가격 :

현재 재고 :

1	2	3
4	5	6
7	8	9
0	delete	enter



4. Spec Review

Important Change

3. 시스템 테스트의 결과(예외처리 보강)로 인한 Exceptional

Use case	15. Add <u>Itemlist</u>
Typical Courses of Events	(AM) : Actor Manager, (S) : System 1. (AM) Window-12(관리자 모드 화면)의 추가버튼을 누른다. 2. (S) Window-13(상품 목록 추가 화면)을 출력한다. 3. (AM) Window-13에서 Item의 name, price를 입력한 후 확인버튼을 누른다. 4. (S) Item 목록을 갱신한다.
Exceptional Courses of Events	1-a. (S) Window-12(관리자 모드 화면)에서 상품 목록 변경의 범위가 지정된 범위(20 종류)를 벗어난다 1-b. (S) 상품 추가 불가 <u>메세지</u> 를 출력한다. 3-(ㄱ) (AM) Window-13(상품 목록 추가 화면)에서 상품명 또는, 가격을 입력안한 채로 확인버튼을 누른다 3-(ㄴ) (S) 잘못된 입력 <u>메세지</u> 를 출력한다. 3-(A) (AM) Window-13(상품 목록 추가 화면)에서 가격에 음수 또는 5만원 초과인 가격을 입력한 채로 확인버튼을 누른다 3-(B) (S) 잘못된 입력 <u>메세지</u> 를 출력한다 3-a. (AM) Window-13(상품 목록 추가 화면)에서 취소버튼을 누른다. 3-b. (S) 관리자 모드 화면을 출력한다.

Use case	17. Change Stock & Price
Actor	Manager
Purpose	상품의 재고를 변경한다.
Overview	상품 재고를 변경한 것을 시스템에 반영시킨다.
Typical Courses of Events	(AM) : Actor Manager, (S) : System 1. (AM) 상품을 선택한다. 2. (S) 해당 상품의 가격&재고 변경 화면을 출력한다. 3. (AM) 상품의 재고와 가격을 변경한 후 확인버튼을 누른다. 4. (S) 변경된 재고와 가격이 지정된 범위 내(재고는 1이상 100 이하, 가격은 0 초과, 50000이하)에 존재한다. 5. (S) 상품의 재고와 가격을 갱신한다.
Alternative Courses of Events	N/A
Exceptional Courses of Events	3-a. (AM) 상품 가격&재고변경 화면에서 취소버튼을 누른다. 3-b. (S) 관리자 모드를 출력한다. 4-a. (S) 변경된 재고와 가격이 지정된 범위(재고는 1 이상 100이하, 가격은 0 초과 50000 이하)를 벗어난다. 4-b. (S) 잘못된 입력 <u>메세지</u> 를 출력한다.

Class
Diagram
Change

JUnit
Test

System
Testing

Spec
Review

Run
Windows
Change

5. Run Windows Change

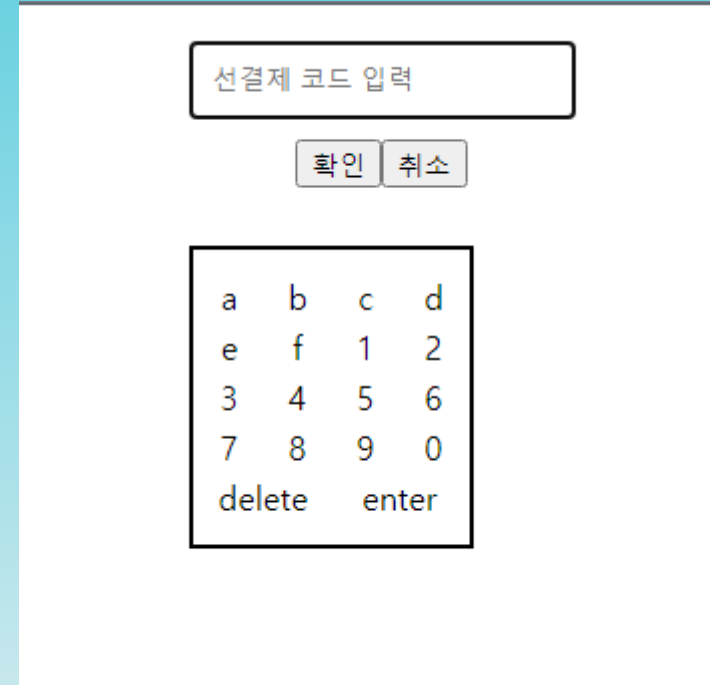
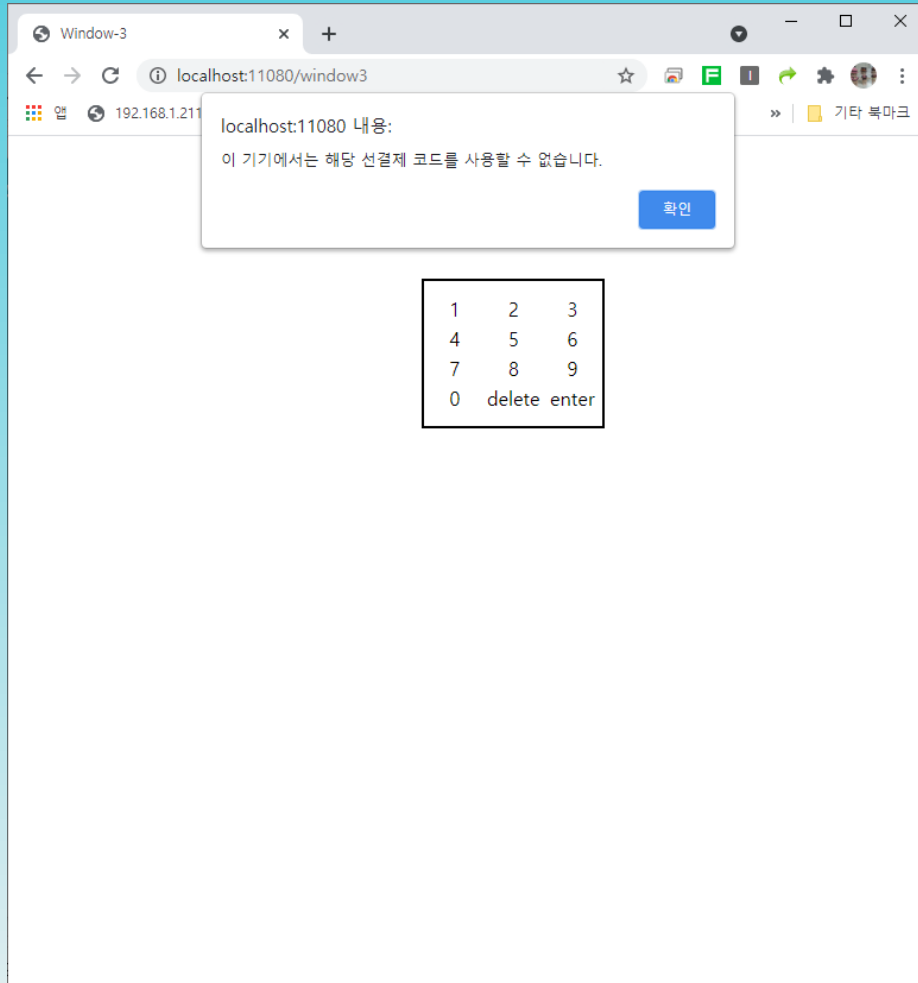
Class
Diagram
Change

JUnit
Test

System
Testing

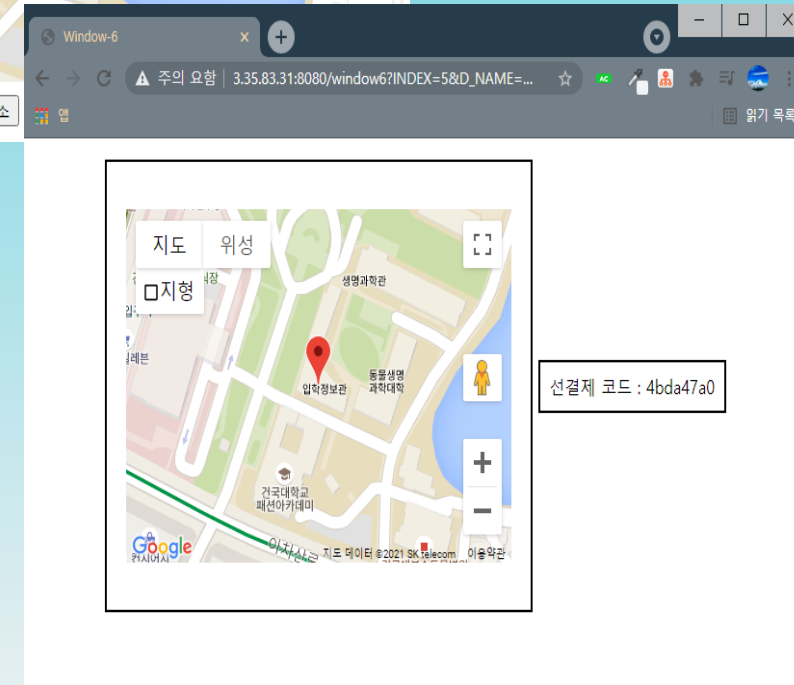
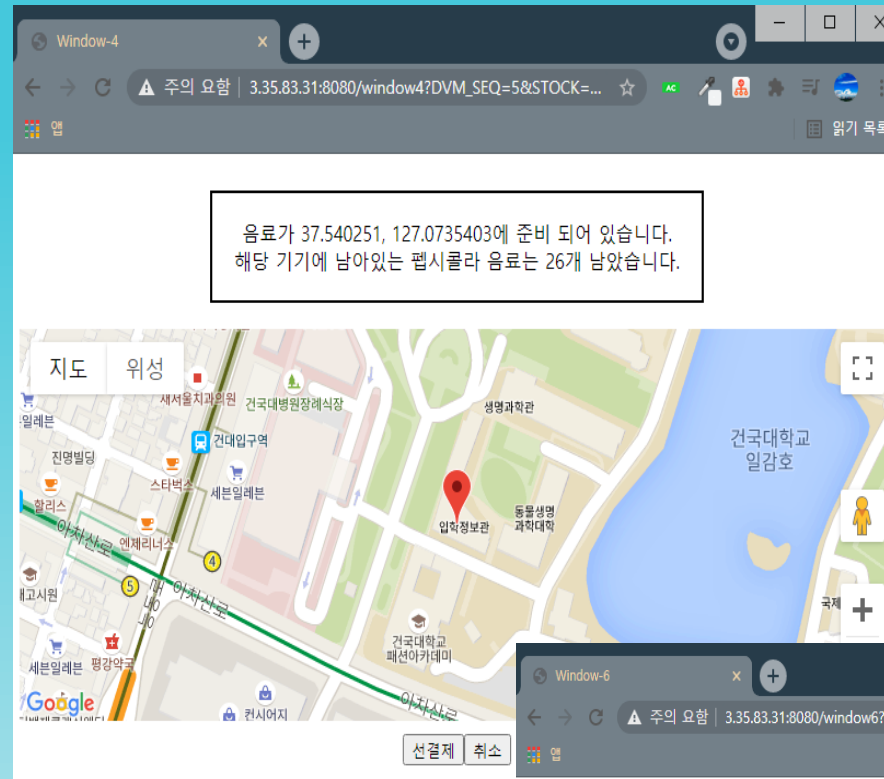
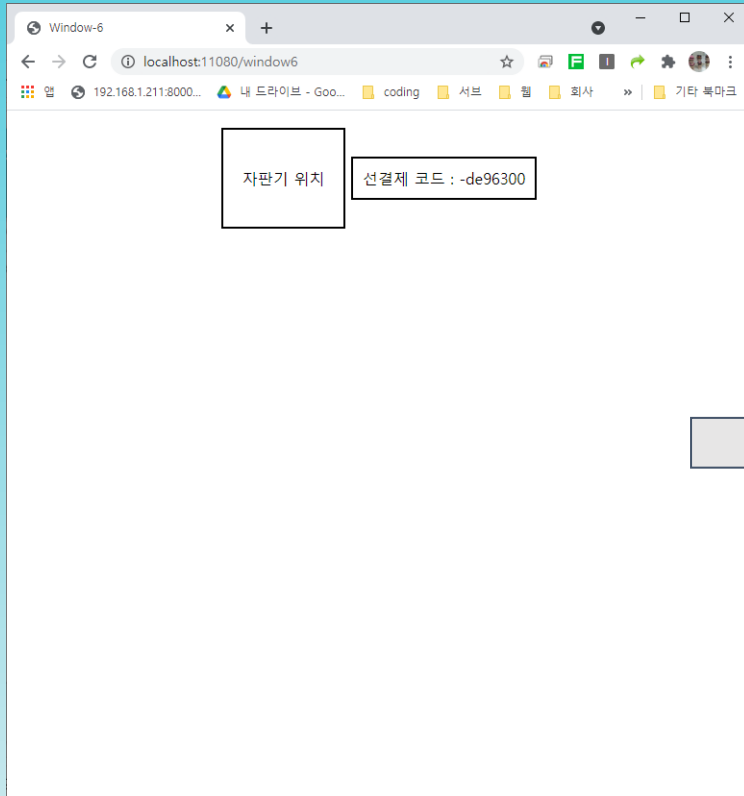
Spec
Review

Run
Windows
Change



5. Run Windows Change

Prepayment



Class
Diagram
Change

JUnit
Test

System
Testing

Spec
Review

Run
Windows
Change

5. Run Windows Change

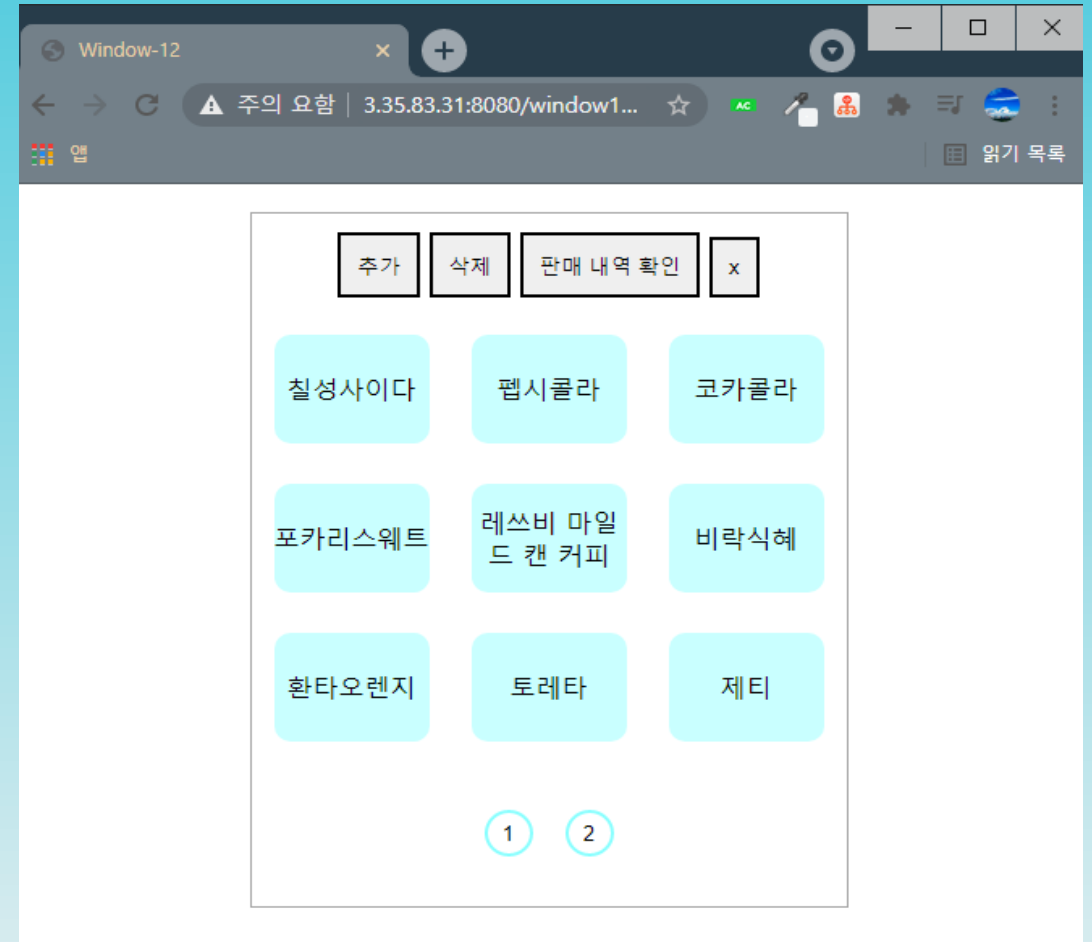
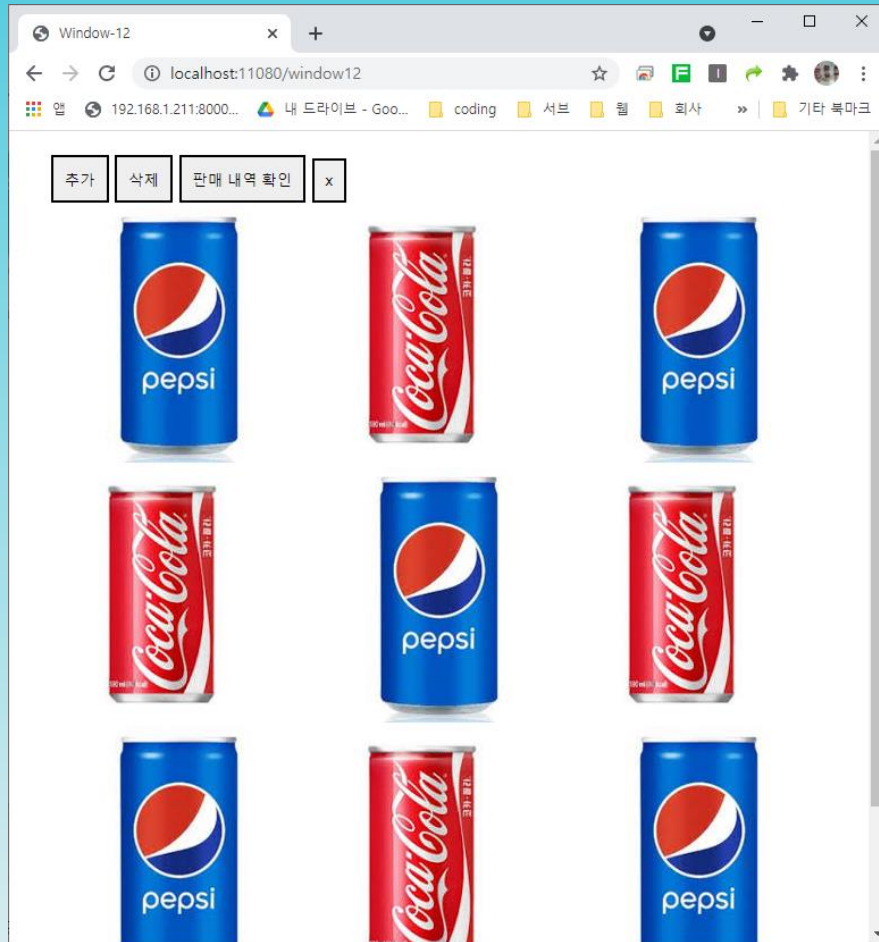
Class
Diagram
Change

JUnit
Test

System
Testing

Spec
Review

Run
Windows
Change



감사합니다