

Distributed vending machine

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

Index

1. Static Code Analysis

2. System Testing

3. Spec Review

4. CTIP환경에 대한 소감

5. UP(OOPT)를 기반으로 SW 개발한 방식에 대한 소감:

실행 파일 링크: DVM: <http://3.35.83.31:8080/>

패킷서버: Packet Server: <http://3.35.83.31:9090/>

☰ 3 Open ✓ 1 Closed	
Static Code Analysis Private 🕒 Updated 12 hours ago	No description
System Testing Private 🕒 Updated 10 hours ago	문서의 내용과 결과물을 기반으로한 테스트 케이스들중 실패한 케이스들에 대한 이슈
Spec Review Private 🕒 Updated 16 hours ago	문서에 대한 이슈

1. Static Code Analysis

Static Code Analysis

System Testing

Spec Review

CTIP

UP

KUSVV-2021 / SMA_Project Private

Code Issues 108 Pull requests Actions Projects 3 Security Insights Settings

Static Code Analysis

Updated 14 hours ago

- To do (15)**
 - [STA 4005] Unused Imports #277 opened by IMRaccoon
 - [STA 4021] Performance - Unused field #293 opened by IMRaccoon
 - [STA 4018] Avoid Static Import #290 opened by IMRaccoon
 - [STA 4002] Abbreviation As Word In Name #274 opened by IMRaccoon
 - [STA 4003] Dataflow Anomaly Analysis #275 opened by IMRaccoon
 - [STA 4016] Visibility Modifier #288 opened by IMRaccoon
- In progress (3)**
 - [STA 1001] Performance – Unused filed #270 opened by IMRaccoon
 - [STA 2001] Member variable visibility should be specified #271 opened by IMRaccoon
 - [STA 3001] Using hardcoded IP addresses is security-sensitive #272 opened by IMRaccoon
- Done (6)**
 - [STA 4015] String literals should not be duplicated #287 opened by IMRaccoon
 - [STA 4007] Synchronized classes Vector, Hashtable, Stack and StringBuffer should not be used #279 opened by IMRaccoon
 - [STA 4008] Variable Declaration Usage Distance #280 opened by IMRaccoon
 - [STA 4017] Avoid Inline Conditionals #289 opened by IMRaccoon
 - [STA 4012] Lines should not be too long #284 opened by IMRaccoon

[STA 2001] Member variable visibility should be specified #271

Opened in KUSVV-2021/SMA_Project

IMRaccoon commented 7 days ago

http://13.209.162.187:9000/project/issues?id=SMA_Project&resolved=false&rules=java%3AS2039

해결방안: 실제 Product 코드에 Logger는 모두 제거

Go to issue for full details

Close issue

1. Static Code Analysis

Static Code Analysis

The screenshot displays the SonarQube web interface for a project named 'SMA_Project'. The 'Issues' tab is active, showing a list of 1 issue. The issue is titled 'Explicitly declare the visibility for "priceMax".' and is categorized as 'Vulnerability'. The issue is located in the file 'src/main/java/Controller/Item.java' at line 15. The code snippet shows the declaration of a static final integer variable 'priceMax' with a value of 50000. A red arrow points to the variable name 'priceMax' in the code. The issue details panel includes a 'See Rule' link, a '3 days ago' timestamp, and a 'L15' severity indicator. The code editor shows the following code:

```
1 package Controller;
2
3 import java.io.BufferedReader;
4 import java.io.IOException;
5 import java.io.InputStreamReader;
6 import java.net.HttpURLConnection;
7 import java.net.URL;
8 import java.nio.charset.StandardCharsets;
9 import java.util.HashMap;
10 import java.util.Map;
11 import java.util.logging.Logger;
12
13 public class Item {
14     private static Logger log = Logger.getLogger(Item.class.getName());
15     static final int priceMax = 50000;
16
17     static String addItemlist(String NAME, int PRICE, int INDEX) throws IOException {
18         if (PRICE > priceMax) {
19             log.info("price is out of range : change price to 50000");
20             PRICE = priceMax;
21         }
22     }
23 }
```

System Testing

Spec Review

CTIP

UP

1. Static Code Analysis

Static Code Analysis

System Testing

Spec Review

CTIP

UP

요약)

Log 가 오류나 특정 상황을 사용자에게 알리기 위한 용도가 아닌

개발에 있어 필요한 상황이라면 배포시에는 제거하는 것이 맞다

-> 해당 로그들을 제거하여 이슈를 해결

[STA 2001] Member variable visibility should be specified #271

Open

IMRaccoon opened this issue 7 days ago · 2 comments



IMRaccoon commented 7 days ago · edited

http://13.209.162.187:9000/project/issues?id=SMA_Project&resolved=false&rules=java%3AS2039

해결방안: 실제 Product 코드에 Logger는 모두 제거



IMRaccoon created this issue from a note in Static Code Analysis (To do) 7 days ago



IMRaccoon added this to the Static Code Analysis (1st cycle) milestone 7 days ago



com8599 commented 3 days ago

차후에 로그를 남길수 있는 상황이 있기때문에 남긴것 입니다. 따로 수정안해도 될까요?



com8599 moved this from To do to In progress in Static Code Analysis 3 days ago



IMRaccoon commented 3 days ago

Author

실제 소프트웨어의 사용과정에서 로그를 남기는 행위가 오류나 특정 상황에 사용자에게 알리기 위한 용도가 아닌 개발에 있어 필요한 상황이라면 배포할 시에는 제거한 뒤에 올리는 것이 맞습니다. 필요하지 않은 내용이라면 제거하는 것이 좋습니다.

1. Static Code Analysis

Static Code Analysis

src/main/java/Controller/Attribute.java

Rename this constant name to match the regular expression `^[A-Z][A-Z0-9]*_[A-Z0-9]*$`. [See Rule](#)

3 days ago L4

Code Smell Critical Open Not assigned 2min effort

convention

```
package Controller;
```

```
public class Attribute {  
    public static final String title = "Title";
```

Rename this constant name to match the regular expression

Code Smell Critical Open Not assigned 2min effort

```
    public static final String data = "Data";
```

Rename this constant name to match the regular expression

Code Smell Critical Open Not assigned 2min effort

```
    public static final String other = "Other";
```

Rename this constant name to match the regular expression

Code Smell Critical Open Not assigned 2min effort

```
    public static final String buy = "Buy";
```

Rename this constant name to match the regular expression

Code Smell Critical Open Not assigned 2min effort

```
    public static final String lo = "LO";
```

System Testing

Spec Review

CTIP

UP

1. Static Code Analysis

Static Code Analysis

Missing curly brace. [See Rule](#)

3 days ago ▾ L132 🔗 ⌵

☠️ Code Smell ⬆️ Critical 🔵 Open Not assigned 2min effort

👉 cert, pitfall

```
StringBuilder postData = new StringBuilder();  
for(Map.Entry<String,Object> param : params.entrySet()) {  
    if(postData.length() != 0) postData.append('&');
```

Missing curly brace. [See Rule](#)

17 days ago ▾ L85 🔗

☠️ Code Smell ⬆️ Critical 🔵 Open Not assigned 2min effort

👉 cert, pitfall

```
    postData.append(param.getKey());  
    postData.append('=');  
    postData.append(param.getValue());  
}
```

System Testing

Spec Review

CTIP

UP

1. Static Code Analysis

코드의 가독성 관련

🔴 [STA 4017] Avoid Inline Conditionals #289

Opened in KUSVV-2021/SMA_Project



IMRaccoon

commented 7 days ago

http://13.209.162.187:9000/project/issues?id=SMA_Project&resolved=false&rules=checkstyle%3Acom.puppcrawl.tools.checkstyle.checks.coding.AvoidInlineConditionalsCheck

해결방안: 삼항연산자 사용을 최대한 배제



🔴 [STA 4012] Lines should not be too long #284

Opened in KUSVV-2021/SMA_Project

해결방안: 가독성을 위해 한 줄에 120자 이상 넘지 않도록 수정

```
$('.round_btn').click( function () {  
    if ( this.classList.value.indexOf("active") >= 0 ) return;  
    let n = this.innerText*1;  
    if ( n-1 != 0 ) n = (9*(n-1));  
    else n = 0;  
    for ( let i = 0+n; i < 9+n; i++ ) {  
        var a = document.getElementById( elementId: (i+1-n)+"");  
        if ( obj[i] != null )  
            a.innerHTML = obj[i].NAME;  
        else  
            a.innerHTML = "";  
    }  
    const btn = document.getElementsByClassName( classNames: "active")[0];  
    btn.classList.remove( tokens: "active");  
    this.classList.add("active");  
});
```


1. Static Code Analysis

Static Code Analysis

System Testing

Spec Review

CTIP

UP

IMRaccoon commented 7 days ago

http://13.209.162.187:9000/project/issues?id=SMA_Project&resolved=false&rules=java%3A51313

해결방안: Environment 를 사용하여 직접 URL이 노출되지 않도록 수정

IMRaccoon created this issue from a note in Static Code Analysis (To do) 7 days ago

IMRaccoon added this to the Static Code Analysis (1st cycle) milestone 7 days ago

com8599 commented 3 days ago

dvm에는 주소줄이 표시되지 않는 환경이기때문에 따로 노출에 대한 보안 이슈를 처리 하지 않는것이 맞는것 같습니다. 문서 수정 드리겠습니다.

요약: 실제 DVM에는 주소표시줄이 뜨지 않는다 그래서 고려할 필요 없다

com8599 moved this from To do to In progress in Static Code Analysis 3 days ago

IMRaccoon commented 3 days ago

해당 내용은 실제 DVM에서 뿐만 아니라 형상관리 내에서도 유관한 내용입니다.
 현재 프로젝트는 Github Private Repository로 운용중이지만 볼 수 있는 사람은 10명정도 됩니다.
 개인 프로젝트에서 Private으로 사용하는 것은 노출되어도 무관하지만
 개인적으로 하는 프로젝트가 아닌 이상 최대한 노출을 하지 않는 것이 좋습니다.
 이와 같은 민감한 주소값들은 한명, 또는 두명 정도의 소수인원만이 해당 정보를 공유하는 것이 바람직합니다.

요약: 개인 프로젝트가 아닌 경우인 이상 주소를 최대한 노출하지 않는 것이 좋다

2.5 Assumptions and dependencies

1. 네트워크 통신 에러는 없다고 가정한다.
2. 각 자판기들의 서로간의 거리는 모두 다르다고 가정한다.
3. 각 자판기의 하드웨어 상의 오류는 없다고 가정한다.
4. DVM은 최대 10대까지 가능하다.
5. 메뉴에서 삭제된 상품의 재고는 관리자가 회수한다.
6. 카드 결제시 잔고를 확인하는 절차가 존재하지 않으므로 일정한 확률(10%)로 결제를 실패시키는 것으로 대체한다.
7. 선결제코드는 선결제 진행 시, 안내받은 자판기에서만 사용가능하다.
8. 판매하는 음료의 총 개수는 최대 20개이다.
9. 각 자판기들에 들어있는 각 상품의 재고는 0개이상 100개이하이다.
10. 자판기에 같은 이름의 상품은 등록할 수 없다.
11. 각 자판기에 판매하는 상품의 종류는 2개이상이다.
12. DVM에는 주소표시줄이 표시되지 않는 환경으로 가정한다.

2. System Testing

Static
Code
Analysis

System
Testing

Spec
Review

CTIP

UP

KUSVV-2021 / SMA_Project Private

Watch 1 Star 0 Fork 0

Code Issues 108 Pull requests Actions Projects 3 Security Insights Settings

System Testing

Updated 10 hours ago

Filter cards Add cards Fullscreen Menu

Column	Count	Item	Details
To do	8	[BF 0712] Add Item 12	#302 opened by IMRaccoon System Testing (2nd cycle)
		[CP 1021] Precode Correct	#208 opened by IMRaccoon System Testing (2nd cycle)
		[BF 0907] Change Stock 07	#304 opened by IMRaccoon System Testing (2nd cycle)
		[BF 0810] Remove Item 10	#303 opened by IMRaccoon System Testing (2nd cycle)
		[BF 0501] Input Precode 01	
In progress	1	[BF 1106] Check Sales History 06	#300 opened by IMRaccoon System Testing (2nd cycle)
Wait for review	1	[BF 0908] Change Stock 08	#305 opened by IMRaccoon System Testing (2nd cycle)
Done	45	[CP 2028] Remove Product 1	#207 opened by IMRaccoon System Testing (2nd cycle)
		[BF 0512] Input Precode 12	#301 opened by IMRaccoon System Testing (2nd cycle)
		[BF 1105] Check Sales History 05	#204 opened by IMRaccoon System Testing (1st cycle)
		[BF 0710] Add Item 10	#184 opened by IMRaccoon System Testing (1st cycle)
		[BF 0709] Add Item 09	

Automated as To do Manage

Automated as Done

2. System Testing

다른 DVM에서 선결제가 이루어진 경우
다른 DVM의 상품의 재고는 감소 하였지만,

판매내역에서 그 상품의 판매 개수가 갱
신되지 않는 문제

← Return to Dashboard
SMA_Project

OVERVIEW TODO MILESTONES TEST RUNS & RESULTS TEST SUITES & CASES REPORTS

T369 [BF 1106] Check Sales History 06 ↓

Brute Force v2 > Brute Force Testing > Check Sales History

Type	Other	Priority	Medium
Automation Type	None		

Preconditions

현재 DVM에 상품이 없어서 다른 DVM에서 선결제가 이루어진 상태

Steps

다른 DVM의 판매 내역 확인 버튼을 누름

Expected Result

선결제가 이루어진 상품의 판매 내역이 갱신됨

RESULTS & COMMENTS ? HISTORY & CONTEXT ? DEFECTS ?

Failed	갱신 되지 않음
2021-06-02 오후 4:04 KUSVV	

Statc
Code
Analysis

System
Testing

Spec
Review

CTIP

UP

2. System Testing

Statc
Code
Analysis

System
Testing

Spec
Review

CTIP

UP

상품 추가 시

상품 이름을 공백으로(아무것도 입력하지 않고) 둘 경우 에러처리 되어있음

하지만 상품 이름 입력 칸에 스페이스바를 통한 빈칸생성 후 (" ") 추가시 상품이 추가 되는 이슈

[BF 0810] Remove Item 10 #303

Open

IMRaccoon opened this issue 7 days ago · 2 comments



IMRaccoon commented 7 days ago

464 : [BF 0810] Remove Item 10

<http://3.35.83.31:8090/testrail/index.php?/tests/view/464>



IMRaccoon added this to the System Testing (2nd cycle) milestone 7 days ago



IMRaccoon added this to To do in System Testing 7 days ago



com8599 commented 3 days ago

상품추가시 빈이름을 입력 불가하게 수정하였기 때문에 이슈가 아닙니다.



com8599 moved this from To do to Wait for review in System Testing 3 days ago



IMRaccoon commented 3 days ago

빈칸을 두칸 넣으면 입력됩니다



IMRaccoon moved this from Wait for review to To do in System Testing 3 days ago

3. Spec Review

Static
Code
Analysis

System
Testing

Spec
Review

CTIP

UP

KUSVV-2021 / SMA_Project Private Watch 1

<> Code Issues 108 Pull requests Actions Projects 3 Security Insights Settings

Spec Review Updated 17 hours ago Filter cards

Column	Count	Item
To do	1	[SR 3018] 2031_07 - 01 #52 opened by jeffkimsh Spec Review (2nd cycle)
In progress	0	
Wait for review	80	[SR 3128] 2033_18 - 01 #267 opened by lji9605 Spec Review (2nd cycle)
		[SR 2102] 2_2.3 - 01 #229 opened by jeffkimsh Spec Review (2nd cycle)
		[SR 3131] 2033_18 - 01 #307 opened by jeffkimsh Spec Review (2nd cycle)
		[SR 3102] 2031_9 - 02 #222 opened by lji9605 Spec Review (2nd cycle)
		[SR 3064] 2033_17 - 01 #93 opened by yuz413 Spec Review (1st cycle)
		[SR 3101] 2031_9 - 01 #219 opened by lji9605 Spec Review (2nd cycle)
Done	111	[SR 3081] 2038_07 - 02 #66 opened by yuz413 Spec Review (1st cycle)
		[SR 3022] 2031_08 - 01 #61 opened by jeffkimsh Spec Review (1st cycle)
		[SR 3032] 2031_11 - 01 #84 opened by jeffkimsh Spec Review (1st cycle)
		[SR 3033] 2031_11 - 02 #85 opened by jeffkimsh Spec Review (1st cycle)
		[SR 3090] 2039 - 01 #81 opened by yuz413 Spec Review (1st cycle)
		[SR 3037] 2031_11 - 06 #94 opened by jeffkimsh

Automated as To do Manage

Automated as Done Manage

3. Spec Review



jeffkimsh commented 22 days ago

- Typical Courses of Events
선결제를 안하는 상황일 때, 카드 잔액이 음료수를 살 수 있을 만큼 충분한 금액이 있는지 확인 하는 것이 없음



ksy0009 commented 5 days ago

잔액확인하는 건 카드리더기(혹은 api)에서 수행하는 부분이고 자판기쪽에선 결제 결과(성공,실패)만 받는 거라 생각했으며, 카드리더기의 구현은 이 과제에서 원하는 방향이 아니라 생각하여 구현하지 않았습니다.
해서 잔액확인에 대한 절차가 없으며 카드 임시 제거, 카드 한도 초과, 잔고부족, 기한 초과 등으로 인한 결제 실패는 확률을 통해 이루어집니다.

-> 신용카드이므로 잔액을 확인하는 과정이 필요없어 구현하지 않았다. 결제는 확률로 성공 또는 실패로 결정하도록 구현하였다.



IMRaccoon commented 4 days ago

저희가 계속 결제 관련되어서 계속 이슈를 만드는 데에는 결제에 대한 확률 계산 방식의 문제입니다.
잔액을 확인하는 것은 카드리더기가 카드를 인식하고, 외부의 API를 사용하여 해당 카드의 유효성 및 잔액 등을 확인하는 것은 알고 있습니다.
하지만 결제 시에 음료수의 값에 따라 결제가 유동적으로 진행되기 때문에 결제 시에 잔액을 확인하는 과정은 필요하다는 것은 알고 계실 겁니다.
다만 90%로 결제가 성공하는 경우, 결제가 성공했는지 실패했는 지에 대한 일관성이 부족합니다.

결제가 가능한 카드일 경우에도 결제가 실패하거나, 결제가 불가능한 카드일 경우에도 결제가 성공할 수 있는 프로그램의 정확성에 대한 문제입니다.
해당 상황의 경우 가장 좋은 방법은 카드의 잔액에 대한 로직을 만들기보다는 카드를 넣는 버튼을 여러가지로 분류하여 구현하고 문서화하는 방식이 존재합니다.
이럴 경우, 잔액이 부족한 경우, 도용된 카드의 경우 등등 다양한 경우에 대해 각각 대처가 가능합니다.
테스트 프로그램이 아니라 실제 문서를 통해 구현하고자 하는 목적이 있는 프로그램에서는 확률이 들어갈 수 없습니다.

-> 그러면 카드의 잔액에 대한 로직을 만들지말고 결제 가능한 카드를 투입, 결제 불가능한 카드 투입 으로 카드 투입(알파벳 'c'입력)을 분기처리해라.

Static
Code
Analysis

System
Testing

Spec
Review

CTIP

UP

3. Spec Review

Important Change

1. 문서들의(Inception, OOA, OOD) 단어 통일

Inception -> 음료

OOA -> 상품

SRS
제품 ->

OOD -> 아이템

-> 상품으로 통일!

[SR 2109] 3_3.2_3.2.2 - 03 #236

Open

jeffkimsh opened this issue 8 days ago · 0 comments



jeffkimsh commented 8 days ago

3.2.2.2 Stimulus/ Response sequence
sequence2의 response에서 '제품의 재고'에서 '제품' 단어 수정



jeffkimsh added this to the Spec Review (2nd cycle) milestone 8 days ago



jeffkimsh added this to To do in Spec Review 8 days ago



easyhooon moved this from To do to In progress in Spec Review 5 days ago



easyhooon moved this from In progress to Wait for review in Spec Review 5 days ago

Static
Code
Analysis

System
Testing

Spec
Review

CTIP

UP

3. Spec Review

Important Change

1. 문서들의(Inception, OOA, OOD) 단어 통일

SRS Stimulus/Response sequence 中

3.2.2 Prepayment

3.2.2.1 Introduction / Purpose of feature

현재 자판기에 재고가 존재하지 않은 상품을 선결제

3.2.2.2 Stimulus / Response sequence

3.2.2.2.1 sequence 1

- stimulus: 카드를 투입(키보드 알파벳 'c' 입력)
- response: 상품 목록 제공

3.2.2.2.2 sequence 2

- stimulus: 사용자가 터치스크린 상의 원하는 상품 버튼을 터치를 통하여 선택
- response: 현 자판기에 선택한 **상품의 재고**가 존재하지 않음을 확인하고 **상품의 재고**가 존재하는 현 자판기로부터 가장 가까운 위치의 자판기의 위치를 안내하고 선 결제 선택 화면을 출력

3.2.2.2.3 sequence 3

- stimulus: 사용자가 선결제를 선택
- response: 사용자가 선택한 상품에 대한 결제를 진행한다.

3.2.2.2.4 sequence 4

- stimulus: 결제 여부 확인
- response: 선결제 코드와 재고가 존재하는 자판기의 위치를 화면에 출력한다.

3.2.2.3 Associated functional requirements

3.2.2.3.1 Functional requirement 1

- 대기 화면이 출력되어야 한다.
- 터치 스크린에 등록되어있는 모든 상품이 출력되어야 한다.

3.2.2.3.2 Functional requirement 2

- 사용자가 선택한 **상품의 재고**가 현재 자판기에 존재하지 않아야 한다.

3.2.2.3.3 Functional requirement 3

- 카드 결제 요청이 승인 되어야한다.
- 선결제 코드를 출력하고 해당 상품의 개수 정보를 업데이트 해야한다.

OOD Design Real Use Cases 中

Use case	10. Serve Item
Actor	System
Purpose	사용자에게 상품 을 배출한다.
Overview	결제된 상품 을 제공한다.
Type	hidden
Cross Reference	System Functions : 2.6 Use Case : "Payment"
Pre-Requisites	상품 에 대한 결제가 되어있어야한다.
Typical Courses of Events	(S) : System <ol style="list-style-type: none"> 1. (S) 결제된 상품을 배출한다. 2. 결제된 상품의 재고를 1개 줄인다 3. (S) 3초동안 <u>Window-8(결제 완료 화면)</u>을 출력한다. 4. (S) Window-2(상품 선택화면)을 출력한다.
Alternative Courses of Events	N/A
Exceptional Courses of Events	N/A

Static
Code
Analysis

System
Testing

Spec
Review

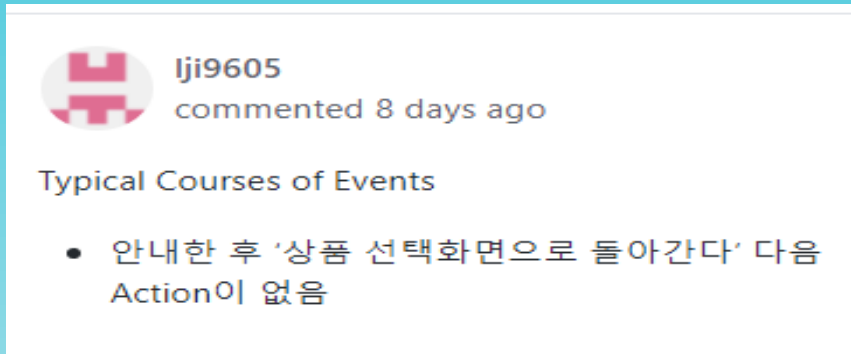
CTIP

UP

3. Spec Review

Important Change

2. 구현 프로그램 내의 함수와 UseCase의 차이점 수정



lji9605 commented 8 days ago

Typical Courses of Events

- 안내한 후 '상품 선택화면으로 돌아간다' 다음 Action이 없음

```
setTimeout( handler: function () {  
    dvm.moveTo2();  
}, , timeout: 3000);
```

구현 시, 편의상 3초가 경과 후 상품 선택화면으로 넘어가는 함수를 UseCase를 구현한 함수에 포함시킴

-> UseCase에도 이를 포함 하도록 구현 -> 문서 수정하는 것으로 결정

변경 결과) OOD Real Use case Diagram 9

Use case	9. Complete Prepayment
Actor	System
Purpose	선결제가 완료되었음을 안내한다.
Overview	선결제 코드와 선결제 코드가 사용 가능한 자판기의 위치를 사용자에게 안내해 선결제를 완료한다.
Type	hidden
Cross Reference	System Functions : 2.1, 2.7 Use Case : "Input Choice", "Create Precode"
Pre-Requisites	재고가 있는 가장 가까운 자판기의 위치 정보와 재고 정보가 현재 자판기에 존재해야 한다.
Typical Courses of Events	(S) : System 1. (S) "Create Precode"에서 생성된 선결제 코드와 사용가능한 자판기 위치를 Window-6(선결제 완료 화면)에 출력한다. 2. (S) Window-2(상품 선택 화면)을 출력한다. 3. (S) "Input Choice"로 넘어간다.
Alternative Courses of Events	N/A
Exceptional Courses of Events	N/A

3. Spec Review

Important Change

3. 소프트웨어적인 표현으로 문서 수정

🔴 [SR 3033] 2031_11 - 02 #85 ×
Opened in KUSVV-2021/SMA_Project

 jeffkimsh
commented 22 days ago

- Overview
상품을 배출한다는 표현은 소프트웨어적 표현
이 아님

🗨️ 0

상품을 배출하는 행위는 구현한 소프트웨어
에선 불가능한 동작

-> 결제 완료 화면 출력으로 대체

선택하신 웰치스 포도맛 나왔습니다

이용해주셔서 감사합니다

3.2. Functional requirements

3.2.1 Direct Payment

3.2.1.1 Introduction / Purpose of feature

현재 자판기에 재고가 있는 상품을 구매

3.2.1.2 Stimulus / Response sequence

3.2.1.2.1 sequence 1

- stimulus: 카드를 투입(키보드 알파벳 'c' 입력)
- response: 상품 목록 화면 출력

3.2.1.2.2 sequence 2

- stimulus: 사용자가 터치스크린 상의 원하는 상품 버튼을 터치를 통하여 선택
- response: 현 자판기에 선택한 상품의 재고가 존재함을 확인하고 결제를 진행한다.

3.2.1.2.3 sequence 3

- stimulus: 결제여부 확인
- response: 결제 완료 화면을 출력

3.2.1.3 Associated functional requirements

3.2.1.3.1 Functional requirement 1

- 대기 화면이 출력되어야 한다.
- 터치 스크린에 등록되어있는 모든 상품이 출력되어야 한다.

3.2.1.3.2 Functional requirement 2

- 사용자가 선택한 상품의 재고가 현재 DVM에 존재해야한다.

3.2.1.3.3 Functional requirement 3

- 카드 결제 요청이 승인 되어야한다.
- 상품 배출한 뒤에 해당 상품의 개수 정보를 업데이트 해야한다.

3. Spec Review

Important Change

그 밖에....

1. 모든 Use case의 대한 Cross reference를 단방향으로 작성한 것을 모두 양방향으로 통일
1. [SR 2104], [SR 3127] 띄어쓰기 수정) 선 결제 -> 선결제
1. [SR 2110] 올바르지 못한 표현 수정) 선결제 상품 입력 -> 선결제 코드 입력
1. [SR 1101] Performance requirements
[주체가 없음) 반응속도가 0.1초 이내여야 한다. -> 터치 스크린의 반응속도가 0.1초 이내여야 한다.

Static
Code
Analysis

System
Testing

Spec
Review

CTIP

UP

4. CTIP 환경에 대한 소감

장점)

1. 깃허브는 단지 code를 management하는 코드 저장소가 아니었다!
 - TestRail, sonarQube 같은 많은 라이브러리와 호환이 가능
 - TestRail, sonarQube를 통해 code의 System testing, Static Analysis 결과를 알 수 있음
 - Code Smell 분석 결과를 통해 내가 구현한 코드의 스타일이 왜 잘못된 것인지, 어떤 스타일로 코드를 짜야되는 것인지 알 수 있음
 - Test case 의 대한 즉각적인 피드백 가능

The screenshot displays the SMA_Project dashboard. The top navigation bar includes 'Return to Dashboard', 'SMA_Project', and 'Working On'. The main menu has 'OVERVIEW', 'TODO', 'MILESTONES', 'TEST RUNS & RESULTS', and 'TEST SUITES & CASES'. The 'Test Runs & Results' section shows two open test runs: 'Brute Force v2' (83% completion) and 'Category Partition v2' (95% completion). Below this, the 'Completed' section shows a date '2021년 6월 3일 목요일' and a version 'BF v1.1'. The 'sonarqube' interface is also visible, showing 'SMA_Project' with 'master' branch. The 'Issues' tab is active, displaying a table of filters:

Type	Count
Bug	1
Vulnerability	1
Code Smell	172
Security Hotspot	10

Severity	Count
Blocker	0
Critical	29
Major	135
Minor	2
Info	8

4. CTIP 환경에 대한 소감

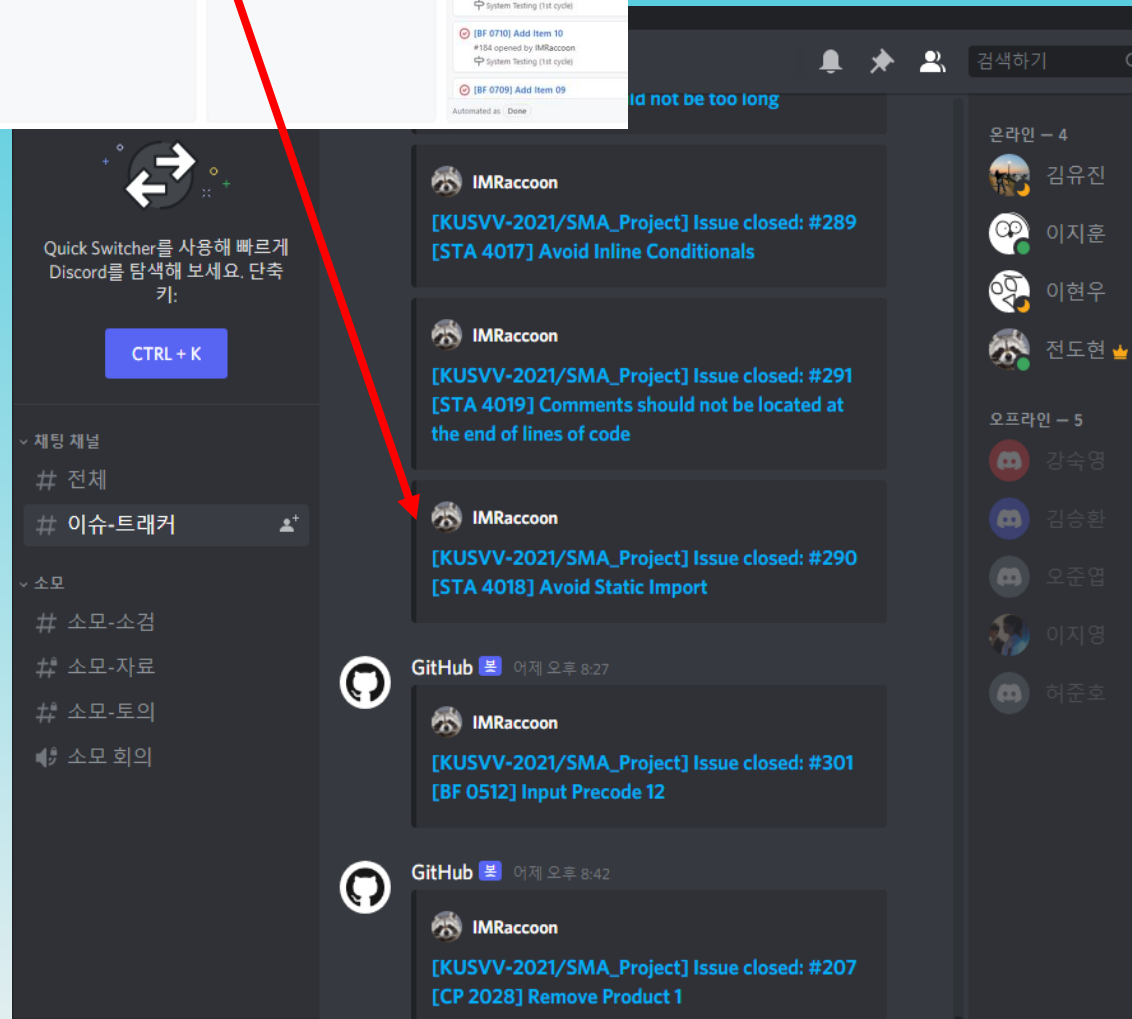
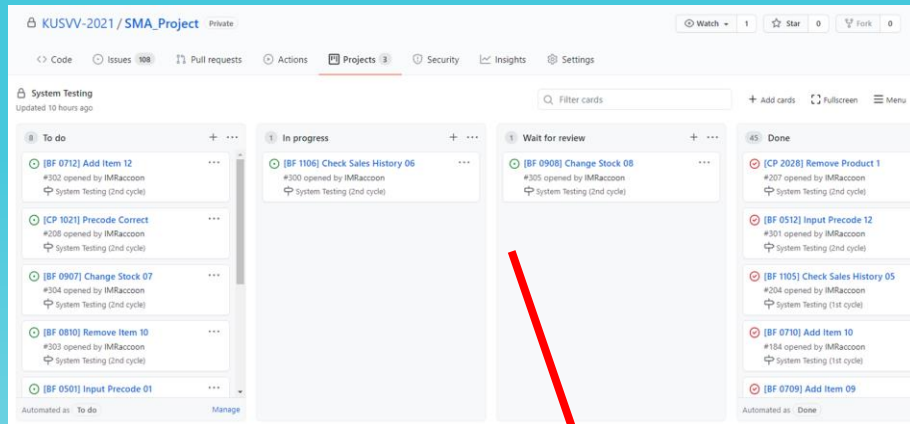
2. 디스코드와 깃허브 연동 (깃허브 봇)

- 실시간으로 각 이슈에 대한 생성 (issue 발생) 리뷰 (issue 의 대한 개발팀의 해결을 검토) 결과 (검토 결과 issue를 종료)

상태를 깃허브에 들어가지 않고도 실시간으로 확인 가능

- 개발 자료와, 각 이슈에 대한 팀의 소통하는 곳을 분리

-> 자료와 회의 결과가 다른 주제의 의견 때문에 묻히지 않음 (<-> 카톡)



Static Code Analysis

System Testing

Spec Review

CTIP

UP

4. CTIP 환경에 대한 소감

Static
Code
Analysis

System
Testing

Spec
Review

CTIP

UP

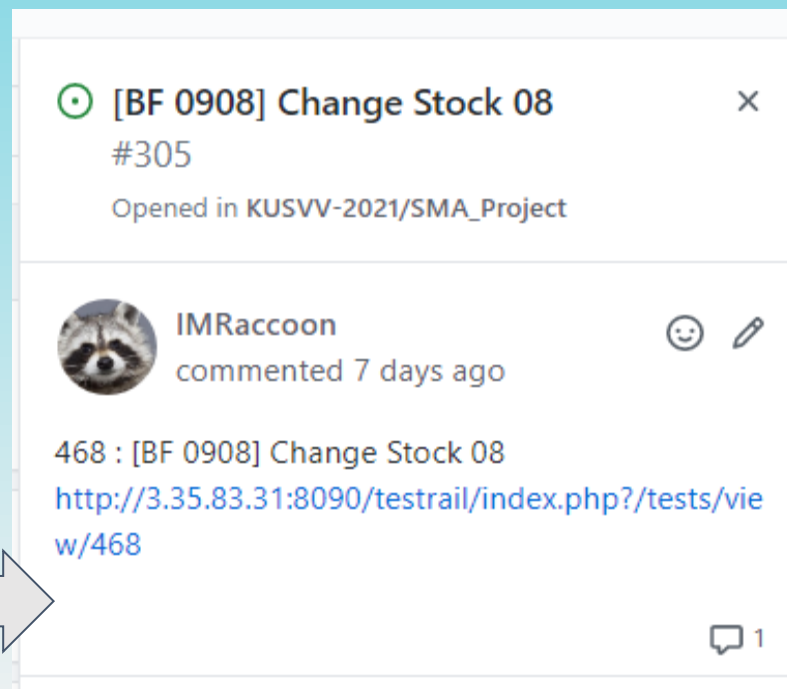
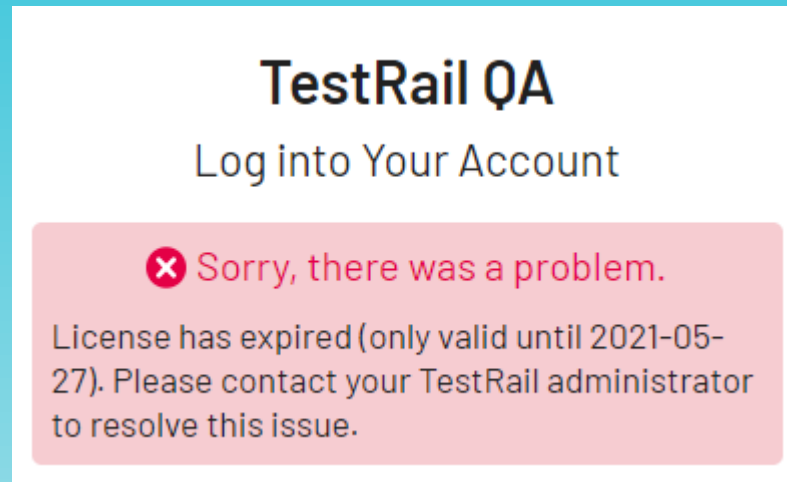
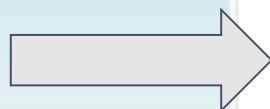
단점)

TestRail의 인증서 기한 문제

개선방향

-> 인증서 기한이 지나도 issue의 내용을 알 수 있도록 github에 issue 내용을 간단히 언급하면 좋겠음

이슈의 내용 언급



5. UP(OOPT) 기반으로 SW개발한 방식에 대한 소감

Static
Code
Analysis

System
Testing

Spec
Review

CTIP

UP

1. 클래스와 클래스내의 함수, Use case의 시퀀스를 먼저 구성한 후 이를 코드화

-> 규모가 큰 프로젝트 일수록 코딩을 하기전에 구조화가 필수적으로 요구

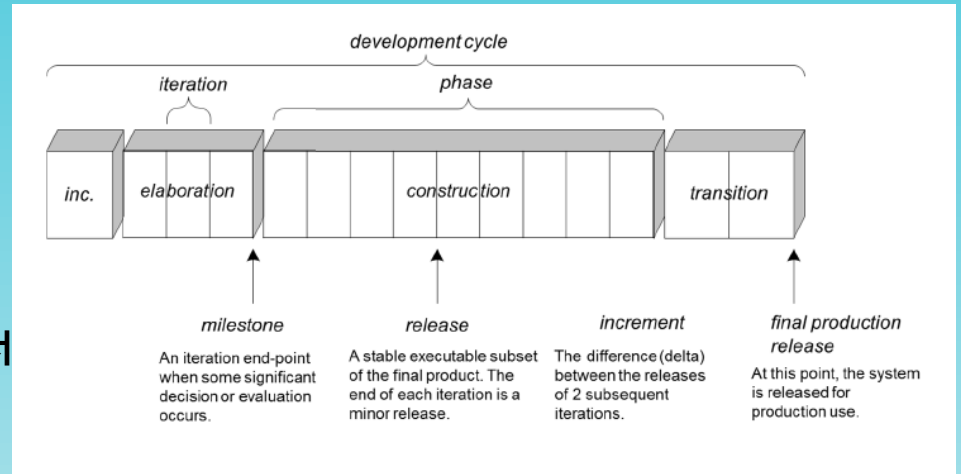
1. 각 Use case에 따른 Alternative, Exceptional course를 미리 문서화

-> 구현 시, 이를 참고하여 예외처리 및 분기 처리를 비교적 쉽고, 체계적으로 관리 가능

1. 사전의 클래스화를 통해 보다 객체 지향적인 개발 가능

-> 각 클래스들의 대한 디자인 패턴 적용 용이

P.S 졸업프로젝트 역시 UP기반 으로 진행되기 때문에 이에 대한 사전적인 경험을 할 수 있었음



OOAD 14p

<http://dslab.konkuk.ac.kr/Class/2021/21SMA/21SMA.htm>

5. UP(OOPT) 기반으로 SW개발한 방식에 대한 소감

Static
Code
Analysis

System
Testing

Spec
Review

CTIP

UP

1. 사이클마다 모든 문서를 수정하는 것에 의한 Risk
 - 하나의 문서에서의 수정을 모든 문서에 반영해야함
 - 문서의 버전관리의 불편함 (Change Log 수기 작성)

1. 2주라는 촉박한 시간내에 구현을 완성해야하는 Risk
 - 시간에 압박감, 구현 실력에 따른 현실적인 구현 방안을 채택
 - 구현과 기존의 계획했던 문서와의 차이 발생
 - 구현 또는 문서 반복적인 수정 작업 필요

개선방향

-> 깃허브와 같은 문서의 수정 후 이 전과의 차이를 한 눈에 비교할 수 있는 툴의 필요성 (Change Log를 수기로 작성해야하는 불편함 개선)

하나의 문서에서 수정이 발생하면 다른 문서도 수정이 자동적으로 지원되는 툴이 필요

구글 공유 문서의 버전 관리의 아쉬움



감사합니다