DISTRIBUTED VENDING MACHINE (DVM)

3rd Cycle development

Project Team: Team 2

Date: 2021.06.11

Team Members :

201310513 황인우 / 201311255 최우석 / 201512265 박인우 / 201711306 박정현

Index

Specification Review 2차 개선 요구사항 반영 결과
Static Analysis 1차 개선 요구사항 반영 결과

Specification Review

2차 개선 요구사항 반영 결과

Failed Specification Review in 1st Cycle

수정 요구사항

17	2030 SRS	9 11	Use Case 14 Type	Typical Courses of Events 에 System 의 Event(Another DVM)만 존재하므로 Hidden 으로 수정 필요	Fail

Specification Review in 2nd Cycle

수정 요구사항

No	Stage	Page	Name	Review
1	1000	7	Requirement – R2.3 Give DVM Info	Type 을 Hidden 으로 수정 필요
2	1000	10	Use Case – R2.3 Give DVM Info	Type 을 Hidden 으로 수정 필요

요구사항 반영 후 문서 내용

: Project Planning Activity 1003, 1006

R2.3	Give DVM Info	Hidden

R2.3	Give DVM Info	14. Give DVM Info	Hidden

Specification Review in 2nd Cycle

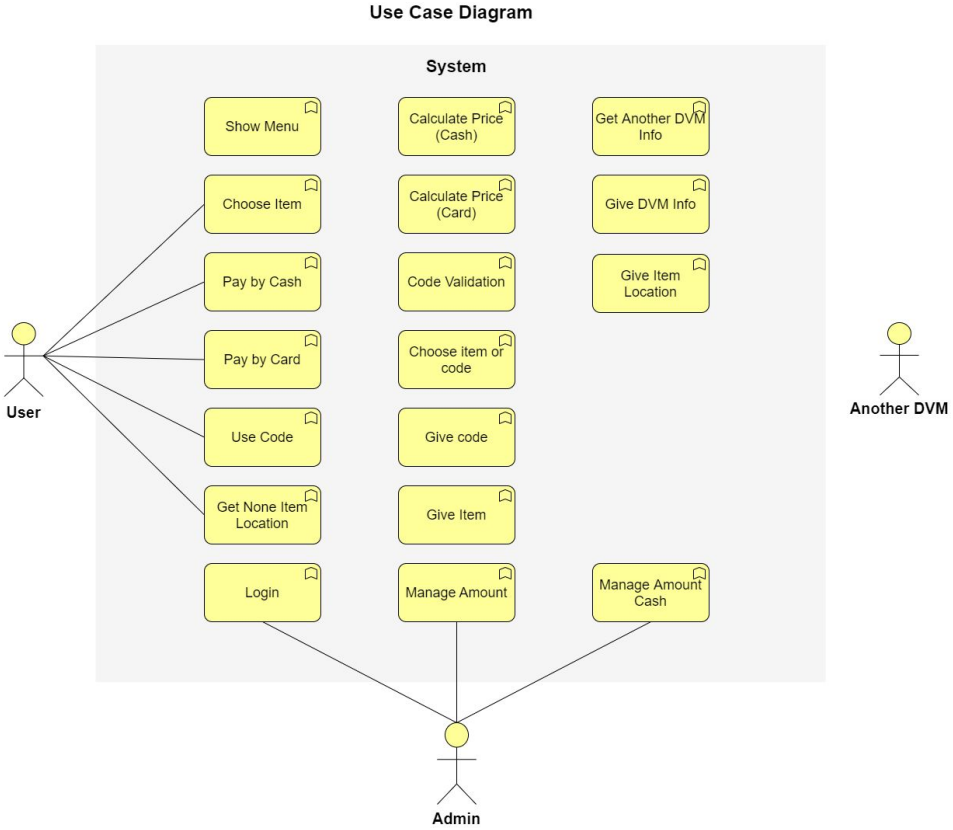
수정 요구사항

3	1000	10	Use Case Diagram	Type 이 Hidden 이므로 System 외부의 Another DVM 으로의 연결 불필요

요구사항 반영 후 문서 내용

: Project Planning, OOA 문서의 Use Case Diagram을 개선했습니다.

Refine Use Case Diagram



Specification Review in 2nd Cycle

수정 요구사항

4	2030	9	Use Case 14 Type	Type 을 Hidden 으로 수정 필요

요구사항 반영 후 문서 내용

: OOA, SRS Use Case 14. Give DVM Info

Use Case	14. Give DVM Info
Actor	Another DVM
Purpose	타 DVM이 정보를 요청할시 현제 DVM의 정보(재고, 위치)를 알립니다
Overview	다른 DVM 에게 현재 DVM의 정보(재고, 위치)를 알립니다
Type	Hidden

Specification Review in 2nd Cycle

수정 요구사항

5	2030	18	System Sequence Diagram – Use Case 14	System Sequence Diagram 불필요

요구사항 반영 후 문서 내용

: OOA 문서의 System Sequence Diagram 중 Use Case 14. Give DVM Info를 삭제했습니다.

Specification Review in 2nd Cycle

수정 요구사항

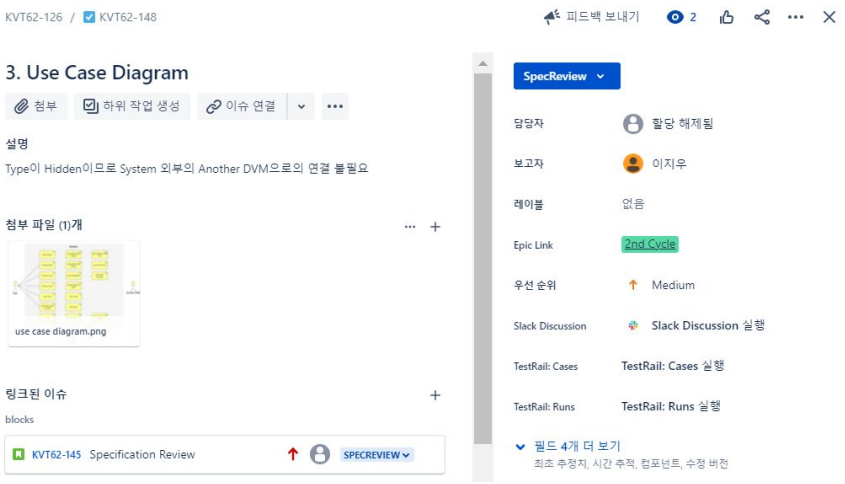
6	2040	10	Use Case 14 Type	SRS 의 Use Case 14 의 Type 은 Hidden 임으로 통일 필요

요구사항 반영 후 문서 내용

: OOD Use Case 14. Give DVM Info

Use Case	14. Give DVM Info
Actor	Another DVM
Purpose	타 DVM이 현 DVM에게 요청한 재고의 유무를 알려줍니다.
Overview	현 DVM에게 정보(해당 상품 재고, 타 DVM 위치)를 반환합니다.
Type	Hidden

Specification Review in 2nd Cycle



2차 Spec Review에 관련된 모든 개선 요구사항들을 해결한 뒤
이슈 관리 칸반보드인 Jira에 해당 수정사항들을 업로드했습니다.

Static Analysis

1차 개선 요구사항 반영 결과

Static Analysis in 1st Cycle

수정 요구사항

1	DVM.java	133	Bug	Minor	Math.pow 함수의 입력은 double형이어 야 합니다.

요구사항 반영 후 소스코드 내용

: DVM.java

```
otherDVM.get(i).setDist(Math.round(Math.sqrt(Math.pow(cur_x - next_x,2.0) +
Math.pow(cur_y-next_y,2.0))*100)/100.0);
```

pow 사용시 double형을 파라미터로 전달하도록 수정 (x,y도 double형으로 수정함)

Static Analysis in 1st Cycle

수정 요구사항

2	DVM.java	26	Code Smell	Major	생성자의 파라미터가 너무 많습니다.

요구사항 반영 후 소스코드 내용

: DVM.java

```
String region, int x, int y, int totalCash, String adminID, String adminPW, ArrayList<Item>
itemList
```

기존 파라미터 개수 총 9개에서

초기값이 0으로 고정된 dist와 기본 Payment로 항상 고정된 myPayment를 삭제

Static Analysis in 1st Cycle

수정 요구사항

3	DVM.java	247	Bug	Major	문자열 비교에 ==을 사용하지 마세요

요구사항 반영 후 소스코드 내용

: DVM.java

```
while (code.equals("000000") || codeTable.keySet().contains(code))
```

기존 코드에서 문자열 비교에 '**==**'를 사용한 부분을 equals로 수정

Static Analysis in 1st Cycle

수정 요구사항

4	DVM.java	29	Code Smell	Critical	String이 반복적으로 나타나고 있습니다.

요구사항 반영 후 소스코드 내용

: DVM.java

```
System.out.println(String.format("[%s]자판기는 생성되지 않았습니다.", region));
```

문자열 포맷팅을 사용해 하나의 프린트문에 여러개의 String을 넣지 않도록 수정

Static Analysis in 1st Cycle

수정 요구사항

5	DVM.java	26	Code Smell	Critical	조건문과 반복문으로 인해 코드가 복잡합 니다.

요구사항 반영 후 소스코드 내용

: DVM.java

```
if (DVMList.size() == 10) {
    System.out.println("자판기는 최대 10개까지만 생성 가능합니다.");
    System.out.println(String.format("[%s]자판기는 생성되지 않았습니다.", region));
    return;
}

if (itemList.size() != 20) {
    System.out.println("아이템리스트는 반드시 20개의 아이템을 보유해야 합니다.");
    System.out.println(String.format("[%s]자판기는 생성되지 않았습니다.", region));
    return;
}
```

기존의 조건문 형태를 하나에서 여러개로 나눠 복잡성을 줄임

Static Analysis in 1st Cycle

수정 요구사항

6	DVM.java	228, 251	Code Smell	Minor	Boolean 대신 boolean 을 사용하세요. Boolean 사용 시 NPE 대비할 수 있도록 코드 작성해야 합니다

요구사항 반영 후 소스코드 내용

: DVM.java

```
void giveItem(Item selectedItem, boolean payByCash)
String giveCode(Item selectedItem, boolean payByCash)
```

Boolean -> boolean 자료형 변경

Static Analysis in 1st Cycle

수정 요구사항

7	Payment.java	13	Bug	Minor	Double Brace Initialization은 권장되지 않 습니다.

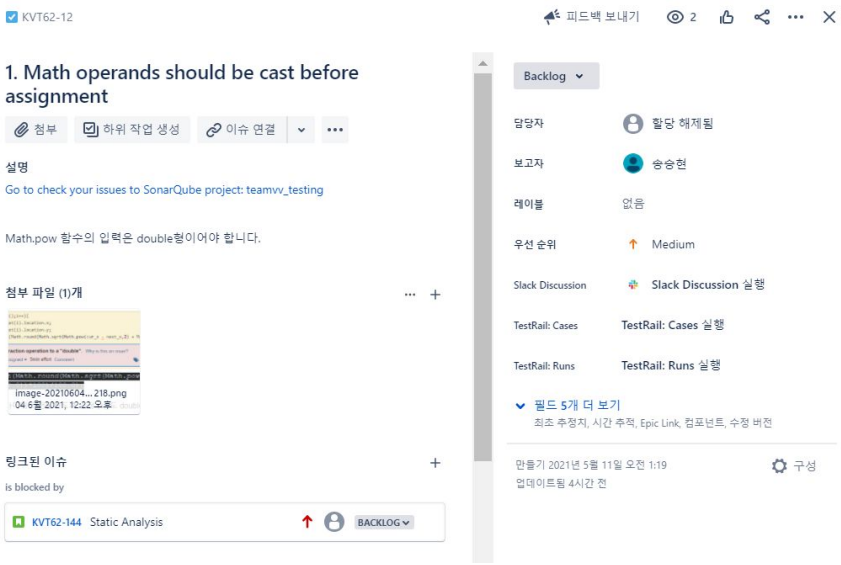
요구사항 반영 후 소스코드 내용

: DVM.java

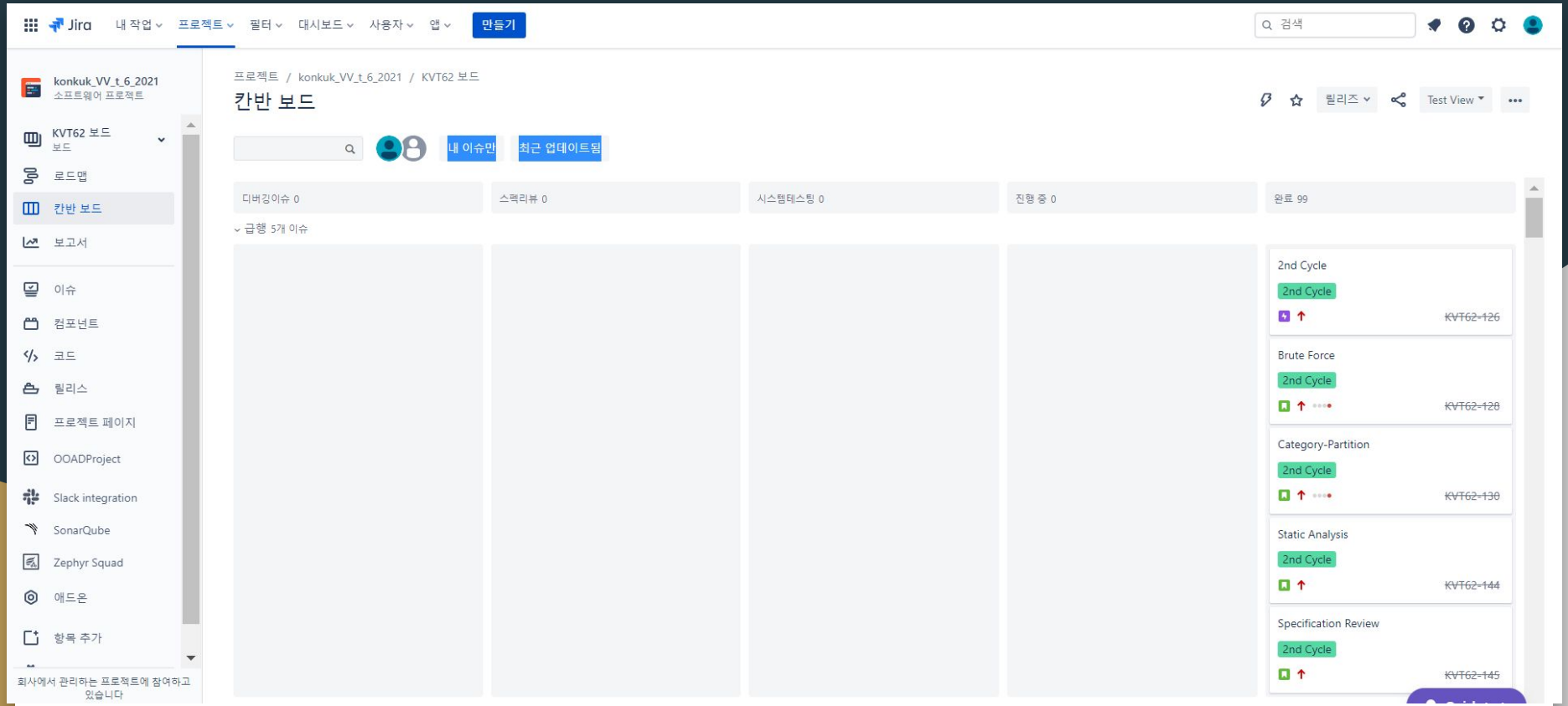
```
public Payment() {
    if (validCardList.isEmpty()) {
        validCardList.put("1234", 999999);
        validCardList.put("5678", 200000);
        validCardList.put("4543", 999999);
        validCardList.put("3572", 1000);
        validCardList.put("1589", 999999);
        validCardList.put("3257", 200000);
        validCardList.put("1088", 999999);
        validCardList.put("0088", 4000);
    }
}
```

한번에 초기화하는 것에서 생성자마다 카드리스트를 확인하고 비어있을 경우 기본값들을 추가하는 방식으로 변경

Static Analysis in 1st Cycle



1차 Static Analysis에 관련된 모든 개선 요구사항들을 해결한 뒤 이슈 관리 칸반보드인 Jira에 해당 수정사항들을 업로드했습니다.



- 검증팀의 확인을 거쳐 검증팀에서 제안한 모든 이슈에 대해 완료 처리했습니다.

Thank You.