

OOI 3rd Cycle

Distributed Vending Machine

TEAM1

202111290 박범식

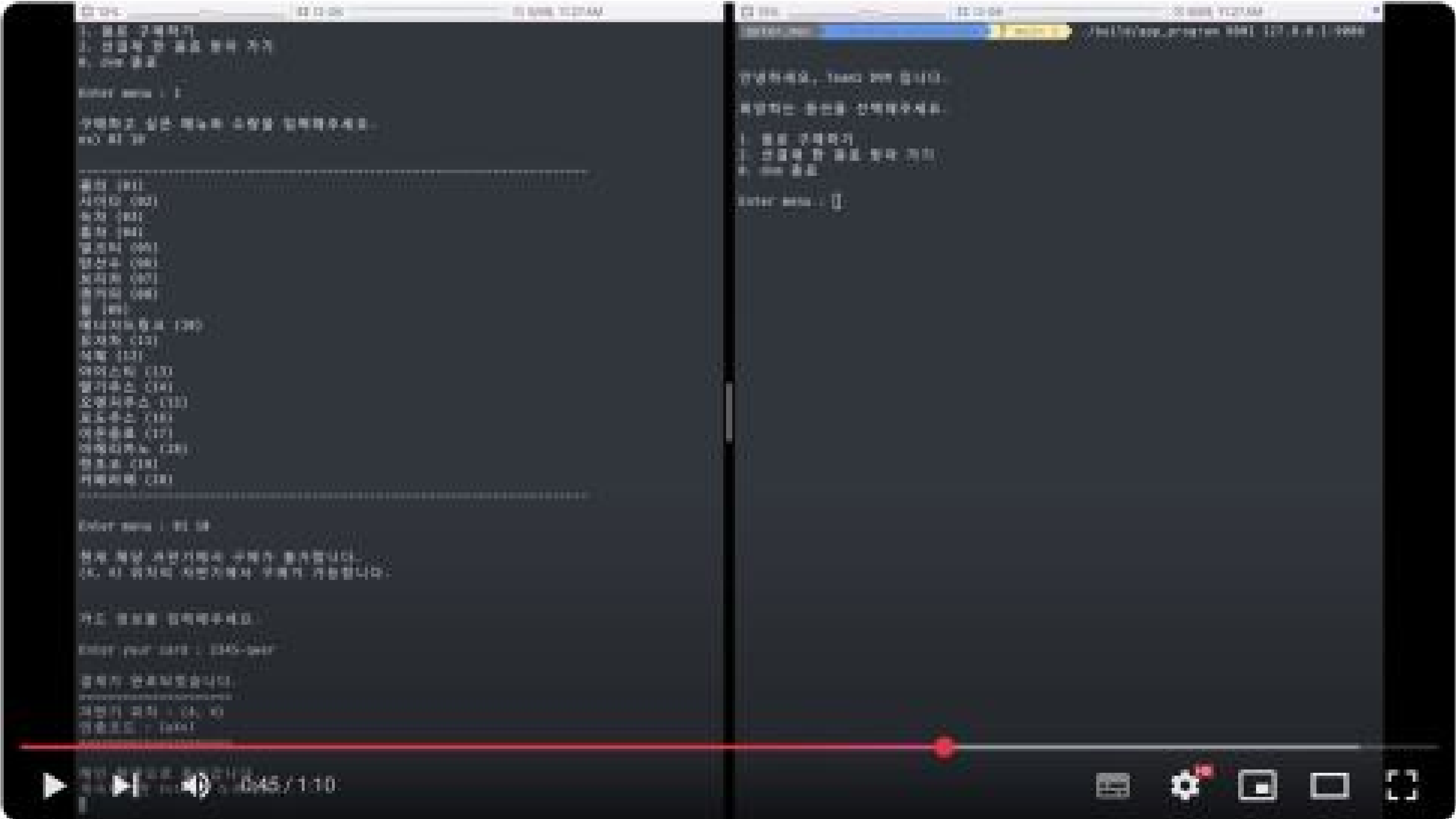
202211382 조휘원

202211373 정지원

Table of Contents

- 1. Demonstration video**
- 2. Construction Review**
- 3. Implementation Review**
- 4. 2nd Cycle Review Issue**
- 5. CI/CD Resolution**
- 6. Impression**

1 Demonstration Video

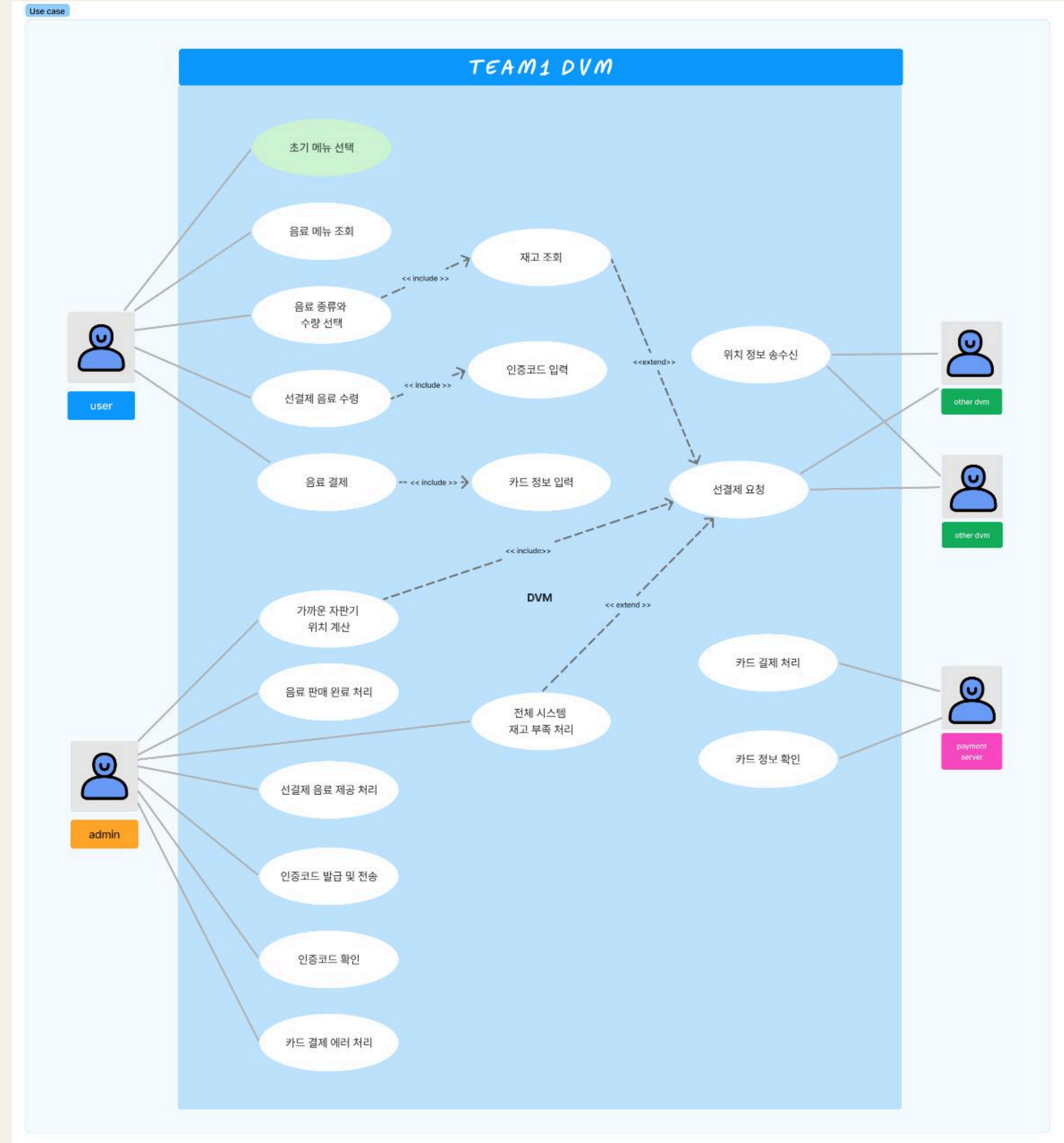


객체지향개발방법론 1조 데모영상

90% 일부 공개

[Youtube Link](#)

01 UseCase



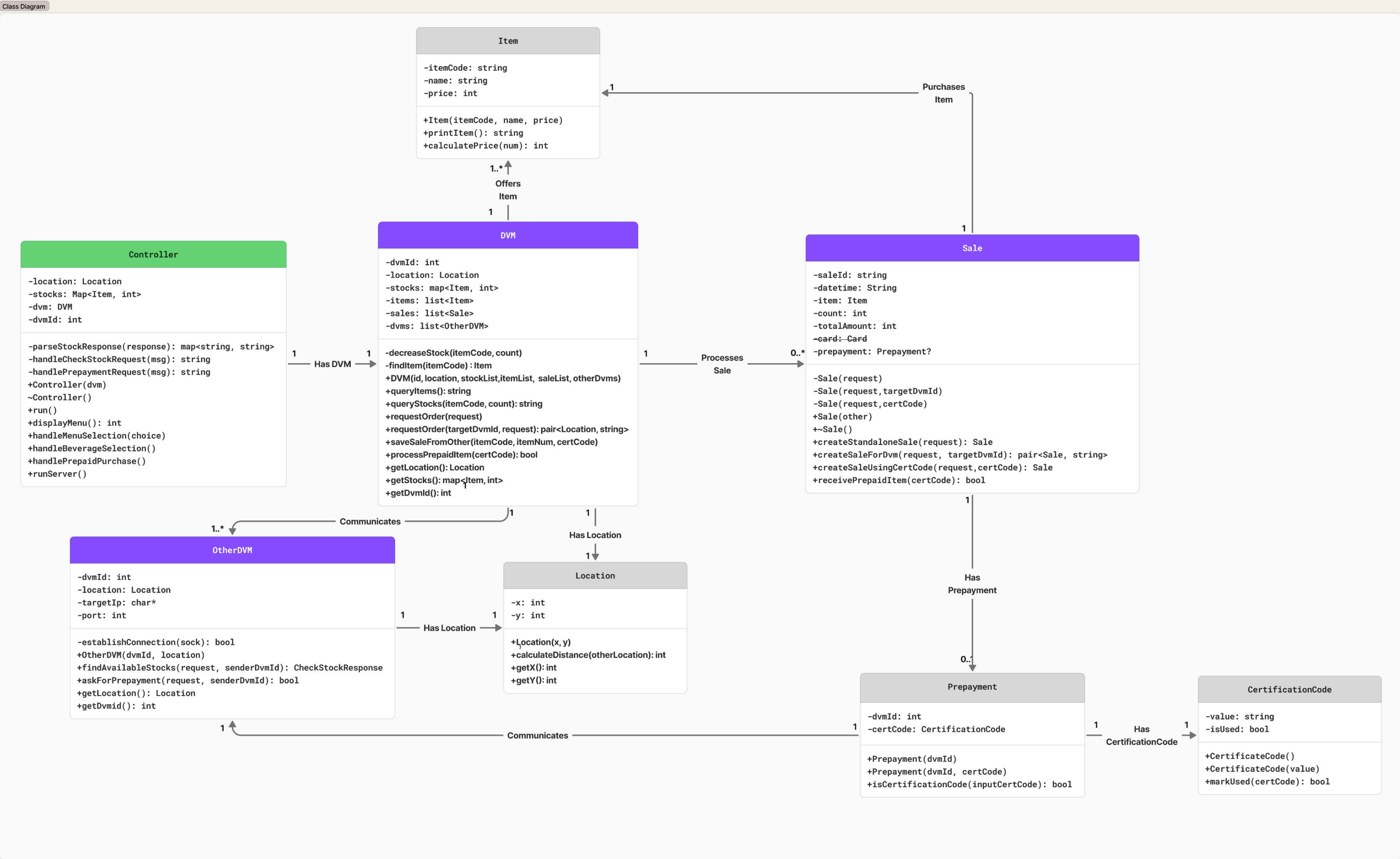
2 Construction Review

02

Design

Class

Diagram



3 Implementaion Review

01
System Test

Category Partition Test

테스트 결과
각 sale_test, domain_test, integration_test, otherdvm_test, controller_test,dvm_test 전부 통과
구현 부분은 다 되어있음 연결만 되면 100%

TestCase16	PASS	정상
TestCase17	PASS	정상
TestCase18	FAIL	다른 DVM 연결 fail
TestCase19	FAIL	다른 DVM 연결 fail
TestCase20	PASS	정상
TestCase21	FAIL	다른 DVM 연결 fail
TestCase22	FAIL	다른 DVM 연결 fail

3/7
42% 달성

4 2nd Cycle Review Issue

01
System Test

Brute Force Test

TestNum	pass/fail	비고
BF-01	pass	음료 구매 화면에서 홈으로 돌아가는 방법이 있으면 좋을 듯
BF-02	pass	
BF-03	pass	
BF-04	pass	
BF-05	pass	
BF-06	pass	
BF-07	pass	
BF-08	pass	수량, 가격 미출력
BF-09	pass	
BF-10	pass	
BF-11	pass	차감이 되는 것 같긴 하지만 육안으로는 확인 불가 (더이상 재고가 없다는 출력보고 임의 확인)
BF-13	pass	
BF-14	pass	
BF-15	pass	
BF-16	pass	
BF-17	pass	
BF-18	pass	
BF-19	pass	
BF-20	pass	
BF-21	pass	
BF-22	pass	
BF-23	pass	
BF-24	pass	
BF-25	pass	
BF-26	pass	
BF-27	pass	
BF-28	pass	

Pass rate = 100%

4 2nd Cycle Review Issue

02

Static Code Analysis



Blocker/High Severity와
주석 제거 문제 위주로 수정

File	Line	Severity	Review
dvm.cpp	60	High	코드에 if for do while switch 문을 3개 이상 중첩하지 않도록 주의필요
	12	Medium	Std::exception, std::logic_error와 같은 일반적인 예외처리를 할려고 하면 std::runtime_error와 같은 예외처리를 잡으려고 하는 모든 코드에 부정적인 영향이 미침
controller.cpp	13	Medium	map<string, string>부분에 const를 선언필요
	112	Medium	필요하지 않는 주석은 제거필요
	359	Medium	전역변수로 e를 사용했는데 사용되지 않는 예외 매개변수 제거필요
sale.cpp	15	High	time(0)가 아닌 nullptr를 사용필요
	16	Blocker	ctime 호출을 ctime_r로 변경 필요
prepayment.h	9	Medium	사용되지 않는 Private field 제거필요
	13	High	prepayment(변환 연산자)에 explicit 키워드가 필요함
otherdvm.cpp	69	Low	If문안에 init문을 사용해서 warnCount를 선언함
Location.h	10	Info	불필요한 주석 제거 필요

전체적으로 const를 사용할 때 비용이 드는 객체들을 사용하여 Medium쪽이 많이 구성됨

4 2nd Cycle Review Issue

02
Static Code Analysis

dvm.cpp

HIGH

if | switch | for | while 등 Code Depth가 3 이상 넘어가면 안됨.



```
51     } else {
52         OtherDVM* nearestDvm = nullptr;
53         int shortestDistance = INT_MAX;
54
55         for (auto& dvm : dvms) {
56             CheckStockRequest request{.item_code = itemCode, .item_num = count};
57             CheckStockResponse response = dvm.findAvailableStocks(request, dvmId);
58             if (response.item_num > 0) {
59                 int distance = location.calculateDistance(dvm.getLocation());
60                 if (distance < shortestDistance) {
61                     shortestDistance = distance;
62                     nearestDvm = &dvm;
63                 }
64             }
65         }
66     }
```

메서드 분리하기
리팩터링으로 조치

4 2nd Cycle Review Issue

02

Static Code Analysis

controller.cpp

MEDIUM

필요하지 않은 주석은 제거 필요



```
109      109      {
110      110          logFile << "[SERVER] Send failed" << std::endl;
111      111      }
112      -      // ::send(client_fd, response.c_str(), response.size(), 0);
113      112      }
114      113
115      114      ::close(client_fd);
```

4 2nd Cycle Review Issue

02

Static Code Analysis

sale.cpp

Blocker

ctime 호출을 ctime_r로 변경 필요



```
time_t now = time(nullptr);  
char buf[26];  
ctime_r(&now, buf);  
datetime = string(buf);
```


4 2nd Cycle Review Issue

02

Static Code Analysis

sale.cpp

HIGH

time(0)가 아닌 nullptr를 사용 필요



```
time_t now = time(nullptr);  
char buf[26];  
ctime_r(&now, buf);  
datetime = string(buf);
```


4 2nd Cycle Review Issue

02
Static Code Analysis

prepayment.h

HIGH

prepayment(변환 연산자)에 explicit 키워드가 필요함



```
12      13      public:
13      -      Prepayment(int dvmId);
14      -      Prepayment(int dvmId, CertificationCode certCode);
15      -      bool isCertificationCode(const std::string& inputCertCode);
16      +      explicit Prepayment(int dvmId);
17      +      explicit Prepayment(int dvmId, CertificationCode certCode);
18      +      bool isCertificationCode(const std::string &inputCertCode);
19      };
20
```

4 2nd Cycle Review Issue

02
Static Code Analysis

Location.h

INFO

불필요한 주석 제거 필요



```
9      10      public:
10      -      // todo : 우리 좌표로 변경 필요
11      11      Location(int x = 0, int y = 0);
12      12      int calculateDistance(Location otherLocation) const;
13      13      int getX() const;
```

SonarQube Analysis

SonarQubecloud

Explore

Search icon

Help icon

Start Free

Log in

advancedsw

Overview

Main Branch

Pull Requests18

Branches1

Information

Collapse

kyunghee > advancedsw > Pull Requests > 20 – 2nd Cycle done

Refresh icon

Success icon

The last analysis has warnings. See details

SummaryIssuesSecurity HotspotsMeasuresCode

PR Summary

151 New Lines • Last analysis 3 days ago • A 4d48c9f8

develop → main

Quality Gate ⓘ

Passed

New Issues0No conditions set

Accepted Issues0Valid issues that were not fixed

CoverageThere is not enough lines to compute coverage

Duplications0.0%Required ≤ 3.0% on 151 New Lines0.0% Estimated after merge

Security Hotspots0No conditions set

6 Impression

기법의 장단점

박범식

설계 단계에서 유즈케이스를 정확히 파악하여 미리 문제를 수정하면 코딩 과정의 수정 작업을 크게 줄일 수 있습니다.
경험이 부족하다면 설계대로 코드를 구현하기 어렵기 때문에 설계 경험이 풍부한 리더가 없다면 효과적이지 않은 개발 방법론이라고 생각했습니다.

조휘원

사전에 충분한 설계를 진행함으로써 개발 초기에는 참조할 기반이 명확해져 수월하게 시작할 수 있었다. 특히 DCD가 많은 도움이 되었습니다. 하지만 설계 단계에서 많은 고민과 높은 이해도를 요구했고, 개발 도중에 설계에서 누락된 부분들을 발견하여면 되돌아가 수정해야 했기에 시간이 많이 드는 기법이라는 점을 체감했습니다.

정지원

설계의 관점에서 다소 시간과 노력이 많이 든다는 점이 유일한 단점이자 치명적인 단점입니다. 이외에는 개발할 때 많은 편의성을 가져다주는 방법론이라고 생각했습니다. 복잡한 시스템을 모듈화하는데 큰 도움을 주었고 코드를 직관적으로 작성하기에 좋은 설계 방법인 것 같습니다.

6 Impression

팀워크

박범식

협업 경험이 풍부한 팀원들과 함께하면서 노하우를 많이 배울 수 있었고, 설계에 대해 팀원들과 협업하며 새로운 관점과 아이디어를 얻을 수 있어서 값진 경험이었습니다. 서로 다른 강점을 가진 팀원들과 함께하며 개인적으로도 많은 성장을 할 수 있었던 것 같아 매우 만족스럽습니다.

조휘원

팀원들이 각자의 역할을 빠짐없이 수행하고, 설계 이해도도 높아 협업 과정이 매우 깔끔하고 효율적이었습니다. 새벽까지 남아 함께 고민하고, 바로바로 질문하며 해결해나가는 과정이 정말 즐거웠습니다.

정지원

4년 간, 팀플 수업을 가장 좋아하고 많이 참여했던 학생으로 살아왔는데 객체지향방법론 수업에서 잊지 못 할 팀원들을 만난 것 같아 너무 행복했습니다. 모든 발표, 모든 과제에서 자신이 맡은 파트 이외에도 열정적으로 참여해준 팀원들 덕분에 좋은 결과로 수업을 마무리할 수 있었습니다.

6 Impression

검증/개발팀의 역할과 책임

박범식

검증팀은 개발팀의 산출물을 사용자 관점에서 철저히 점검하고, 상세한 검증 결과를 개발팀에 전달해야 합니다.
개발팀은 이를 수용하여 산출물 개선에 반영하는 명확한 절차가 필요하며, 두 팀 간의 적정 소통 수준에 대한 검토도 함께 이루어져야 합니다.

조휘원

개발 팀의 입장에서, 검증 팀에게 실행 방법과 설계 내용을 명확하고 친절하게 전달하는 것이 중요하다는 걸 느꼈고, 검증 팀 역시 충분한 이해를 바탕으로 실시간 소통이 가능한 환경을 갖추는 것이 필수적임을 느꼈습니다.

정지원

개발팀은 설계 내용을 토대로 로직 관점에서 좋은 프로덕트가 나오도록 책임을 다해야 합니다. 검증팀은 개발팀을 뒷받침하며 미처 확인하지 못 했던 오류 사항을 검토할 수 있는 검증 도구 및 테스트 계획을 제대로 수립하는 책임을 다해야 합니다. 두 책임은 계속 긴밀히 협력하여 지속적으로 함께 작업해야 하기에 하나의 프로덕트에 대해 모두의 이해도가 뒷받침 되어야 합니다.

Thank you!