

Distributed vending machine

OOPT Stage 1000 - Plan (OOPT_ALL)

Team 4

곽민정 김재윤 이승진 최준

Contents

1. Activity 1001. Define Draft plan
2. Activity 1002. Create Preliminary Investigation Plan
3. Activity 1003. Define Requirements
4. Activity 1004. Record Terms in Glossary
5. Activity 1005. Implement Prototype (pass)
6. Activity 1006. Define Business Use Case
7. Activity 1007. Define Business Concept Model
8. Activity 1008. Define Draft System Architecture
9. Activity 1009. Define System Test Case

Activity 1001. Define Draft plan

Motivation: 기존 자판기의 문제점 개선

1. (관리자) 재고 파악 및 관리가 어렵다.
2. (사용자) 자판기에 재고가 없다면 먹을 수 없거나 다른 자판기를 찾아야 한다.
3. (사용자) 다양한 자판기가 있지만 원하는 음료가 어디에 있는지 파악하기 어렵다.

Activity 1001. Define Draft plan

Project Objectives: 기존 자판기의 불편함 해소 및 사용성 증대

1. 자판기들을 서로 연결 → 재고 파악과 고장 등에 유리한 시스템
2. 재고가 부족하거나 사용자가 원하는 음료가 없는 경우, 다른 자판기와의 통신을 통해 해당 음료를 구입할 수 있는 다른 자판기를 안내 → 사용자가 겪는 불편 감소, 음료 선택의 폭 확대
3. 사용자가 다른 자판기에서의 구매를 선택하는 경우, 선결제 서비스 제공 → 구매 절차를 간소화

Activity 1001. Define Draft plan

Functional Requirements

1. Payment : 결제
2. Inform Location : 자판기 위치 정보 제공
3. Advance Payment Verification Code : 선결제 인증코드 처리
4. Network Message : 네트워크 Message 처리
5. Stock Update : 자판기 재고 최신화
6. Vending Item : 음료 제공
7. View Item : 제공하는 음료의 종류 확인

Activity 1001. Define Draft plan

Non-Functional Requirements

1. 기능성: 재고관리가 잘 되어야 한다.
2. 신뢰성: 고장이 났을 때 빠르게 고쳐져야 한다.
3. 무결성: 모든 요청에 대해 데이터의 정확성과 완전성을 보장한다.
4. 성능: 통신 속도가 빨라야 한다. (Latency가 없어야 한다)
5. 사용성: 자판기 사용이 편리해야 한다.
6. 사용성: 사용자와 시스템간 소통을 간단하게 하여, 사용자의 잘못된 조작이 시스템 동작을 방해할 여지를 최소화한다.
7. etc: Java 사용

Activity 1002.

Create Preliminary Investigation Plan

Alternative Solutions

1. 마트/편의점
2. 카페/오프라인 매장
3. 온라인 쇼핑몰

Project Justification (Business needs)

1. 높아진 땅값으로 인해 오프라인 매장의 부지 확보가 어려움
2. 코로나 사태로 인해 비대면 서비스의 수요가 증가
3. 배달비/운송료 인상으로 인한 배송 서비스의 비용 증가

Activity 1002.

Create Preliminary Investigation Plan

Business Market

1. 코로나 사태로 인한 비대면 서비스 수요 증가 → 간단한 상품들을 판매하는 자판기 필요
2. 음료 뿐만 아니라 다양한 상품들이 자판기로 판매되고 있음
3. 재고 부족이나 잦은 고장 등의 이유로 외면 받던 기존 자판기에 기능을 추가하여 재사용 가능

Managerial Issues

1. 2개월 내 개발 완료

Activity 1002.

Create Preliminary Investigation Plan

Risks Management & Risks Reduction

1. 코로나로 인한 대면의 어려움 → ZOOM, Discord 및 여러 도구를 활용하여 온라인 회의 진행
2. 코로나로 인한 건강 악화 위험 → 개인 건강관리에 유의
3. 타 수강 과목으로 인한 부담 → 개인 과제와 강의 성실히
4. 개발능력의 부족 → 문제 사항의 활발한 소통을 통해 신속하게 해결

Activity 1003. Define Requirements

Define Functional Requirements

1. Payment : 결제

1. 사용자는 원하는 음료를 선택하여 카드로 결제할 수 있다.
2. 입력한 카드번호가 유효한지 확인한다.
3. 카드번호가 유효하지 않거나 잔액이 부족한 경우 결제를 취소한다.
4. 결제하는 도중 재고가 떨어진 경우 환불해준다.
5. 사용자는 다른 자판기의 음료 구매를 위해 선결제를 할 수 있다.

Ref	Function	Category	Ref	Function	Category
R 1.1	selectItem	Evident	R 1.5	cancelPayment	Hidden
R 1.2	isValidCard	Hidden	R 1.6	refund	Hidden
R 1.3	isEnoughBalance	Hidden	R 1.7	prePay	Evident
R 1.4	buyItem	Evident			

Activity 1003. Define Requirements

Define Functional Requirements

2. Inform Location : 자판기 위치 정보 제공

1. 관리자가 각각의 DVM에 대하여 id와 위치정보(좌표)를 설정한다.
2. 사용자가 현재 자판기에서 팔지 않거나 재고가 없는 음료를 선택한 경우, 해당 음료를 구매할 수 있는 가장 가까운 위치의 자판기를 안내 받을 수 있다.

Ref	Function	Category
R 2.1	setDVMInfo	Evident
R 2.2	getNearestDVM	Hidden

Activity 1003. Define Requirements

Define Functional Requirements

3. Advance Payment Verification Code : 선결제 인증코드 처리

1. 사용자가 선결제를 요청하는 경우 현재 자판기에서 선결제 후 인증 코드를 발급한다.
2. 사용자가 다른 자판기로 가서 인증 코드를 입력하면 음료가 나온다.

Ref	Function	Category
R 3.1	createCode	Hidden
R 3.2	isVaildCode	Hidden

Activity 1003. Define Requirements

Define Functional Requirements

4. Network Message : 네트워크 Message 처리

1. Request Message
2. Response Message
3. DVM 간 통신은 오른쪽 표와 같은 규칙을 따른다.

Ref	Function	Category
R 4.1	sendReqMsg	Hidden
R 4.2	sendResMsg	Hidden

src id	dst id	msg type	msg description	msg direction
현재 자판기 id	대상 자판기 id	msg 타입	msg 내용	msg 방향
		재고 확인 요청	음료 코드_음료 개수	Broadcast
		재고 확인 응답	음료 코드_음료 개수_dst id_dst 좌표	dst→src
		선결제 확인	음료 코드_음료 개수_인증 코드	src→dst
		음료 판매 확인	음료 코드_음료 개수	broadcast
		음료 판매 응답	음료 코드_dst id_dst 좌표	dst→src

Activity 1003. Define Requirements

Define Functional Requirements

5. Stock Update : 자판기 재고 최신화

1. 사용자가 결제 완료한 경우, 결제 완료한 만큼 재고를 차감한다.
2. 관리자가 재고를 보충한 뒤, 재고관리 모드에 진입하여 해당 자판기의 재고 상태를 업데이트 해준다.

Ref	Function	Category
R 5.1	subtractStock	Hidden
R 5.2	updateStock	Evident

Activity 1003. Define Requirements

Define Functional Requirements

6. Vending Item : 음료 제공

1. 총 음료는 20 종류다.
2. 한 자판기당 7종류의 랜덤한 음료 판매한다.
3. 사용자가 결제한 상품을 배출한다.

Ref	Function	Category
R 6.1	provideItem	Hidden
R 6.2	setTotalItemList	Evident
R 6.3	setItemList	Evident

Activity 1003. Define Requirements

Define Functional Requirements

7. View Item:

1. DVM에서 판매하는 20종류 음료의 이름과 가격을 표기한다.

Ref	Function	Category
R 7.1	showItemList	Hidden

Activity 1003. Define Requirements

Define Non Functional Requirements

1. 성능

- 자판기간의 통신 응답 시간은 1초 이내로 수행되어야 한다.
- 데이터를 확인하는 시간은 1초 이내로 수행되어야 한다.

2. 사용성

- 친화적이고 직관적인 인터페이스를 제공한다.

3. 기능성

- 지속적 모니터링을 통해 신속한 재고관리를 한다.

Activity 1003. Define Requirements

Define Non Functional Requirements

4. 신뢰성

- 고장이 났을 때 빠르게 고칠 수 있도록 미리 위험요소를 분석하고 지속적으로 모니터링 한다.

5. 무결성

- 자판기 사이의 msg protocol 을 준수한다.

Activity 1003. Define Requirements

Other Requirements

1. Language: Java
2. Operating Environment: JVM and Windows 10

Activity 1004.

Record Terms in Glossary

Record Terms in Glossary

terms	description
DVM	Distributed Vending Machine의 약자. 네트워크로 연결된 각각의 자판기를 지칭한다.
system	네트워크로 연결된 전체 DVM 시스템을 지칭한다.
결제	DVM에 음료의 값 만큼 카드로 값을 지불하는 것이다.
선결제	다른 DVM에서 음료를 받기 위해 현재 DVM에서 먼저 결제하는 것이다.
재고	각 DVM에서 가지고 있는 음료의 종류와 개수이다. 각 음료의 재고는 최대 999개이다.
인증 코드	알파벳 소문자와 숫자를 모두 포함하는 10자리 문자열. 사용자가 선결제를 완료한 경우에, 현재 DVM에서 발급하는 코드이다. 다른 DVM에서 인증코드 입력 시 선결제한 음료가 나온다.
좌표	각 DVM의 위치를 나타내는 문자열이다. (x,y)형태이며 x, y의 범위는 0~99이다. 각 DVM의 좌표는 서로 겹치지 않는다.
id	각 DVM의 재고와 거리, 위치를 파악하기 위한 고유 번호이다. Team+n의 형태이다.

Activity 1004.

Record Terms in Glossary

Record Terms in Glossary

terms	description
위치 정보	DVM들의 좌표 정보로 계산한 각 DVM의 위치이다.
request message	DVM이 다른 DVM에 음료 코드, 재고를 확인할 때, 인증 코드 보낼 때 네트워크를 통해 보내는 메시지이다. 데이터 타입은 String이고 음료 개수는 10진수 세 자리 숫자로 표기하며, 데이터는 '_'를 사용해 구분한다.
response message	DVM에서 음료 재고가 있는 경우 DVM에게 id와 좌표를 네트워크로 통해 보내는 메시지이다. 데이터 타입은 String이고 음료 개수는 10진수 세 자리 숫자로 표기하며, 데이터는 '_'를 사용해 구분한다.
item	자판기에서 제공하는 음료를 뜻한다.
사용자(user)	DVM에서 음료를 구매하려고 하는 사람을 뜻한다.
관리자(admin)	각각의 DVM을 관리하는 사람. 해당 자판기의 재고상태를 업데이트 한다.

Activity 1006.

Define Business Use Case

Identify user goals for each actor

1. Primary Actors

1. 사용자

1. DVM이 제공하는 음료의 종류를 확인한 뒤 구매
2. 해당 DVM에 재고가 없을 경우, 재고가 있는 다른 DVM 위치를 안내 받음
3. 안내 받은 DVM에 대한 선결제

2. 관리자

1. DVM 정보 세팅 및 재고 보충, DVM 갱신

2. Supporting Actors

1. Other DVMs

1. 다른 DVM과 통신하여 재고 여부 파악
2. 타 DVM의 선결제 요청에 따른 재고 확보 및 인증코드 관리

Activity 1006.

Define Business Use Case

Categorize identified use cases

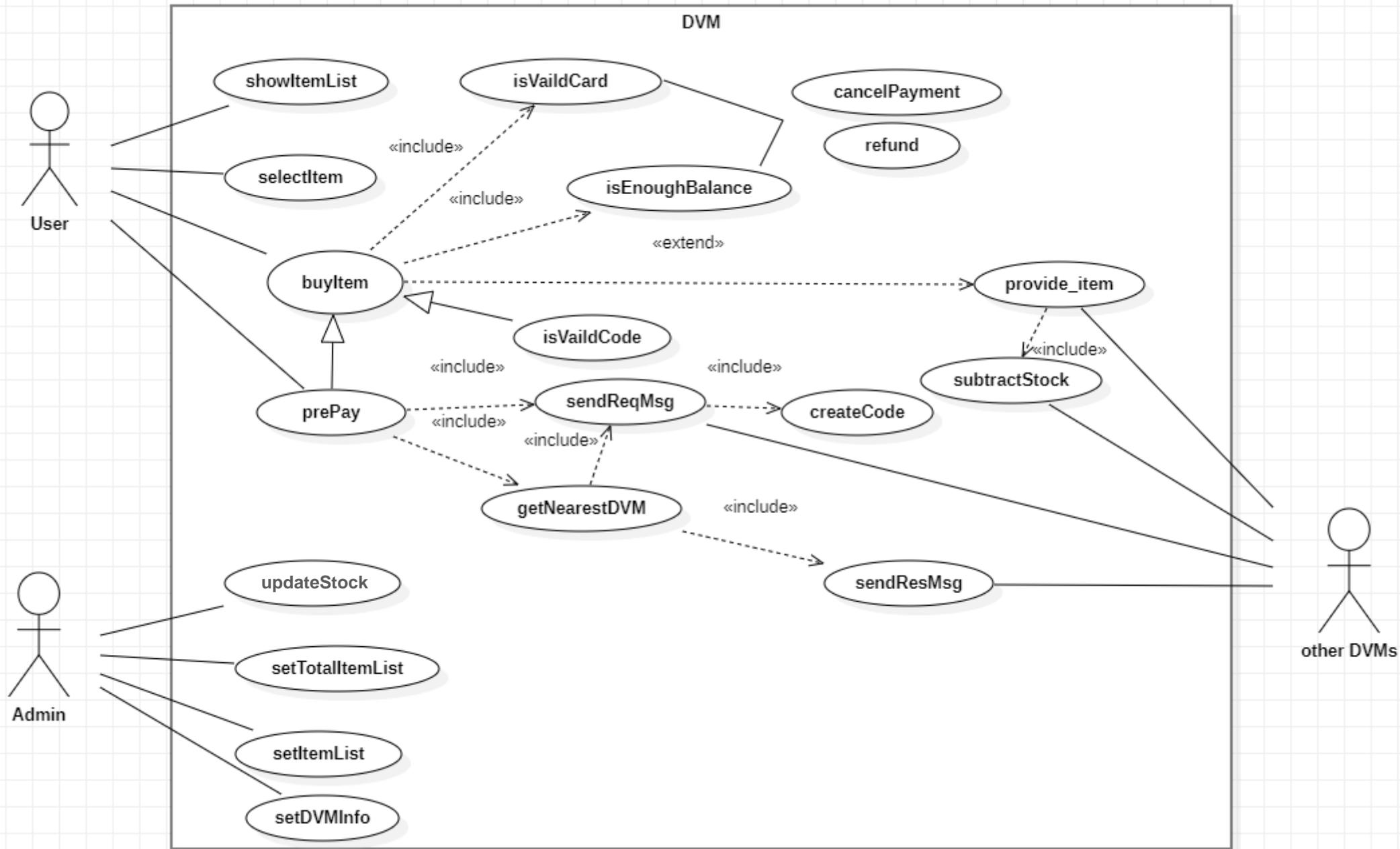
1. primary use cases
 - selectItem, buyItem, prePay, sendReqMsg, sendResMsg, subtractStock
2. secondary use cases
 - createCode, isVaildCode, isValidCard, cancelPayment, getNearestDVM, refund, provideItem, isEnoughBalance, setTotalItemList, setItemList, showItemList, setDVMInfo
3. optional use cases
 - updateStock

Activity 1006.

Define Business Use Case

Identify relationships between use cases

(다음 슬라이드)



Activity 1006.

Define Business Use Case

Describe use cases

Use Case	selectItem
Actor	사용자
Description	사용자가 음료를 선택한다.

Use Case	buyItem
Actor	사용자
Description	선택한 음료에 대해 결제를 수행한다.

Use Case	isVaildCard
Actor	None
Description	사용자의 카드번호가 유효한지 확인한다

Activity 1006.

Define Business Use Case

Describe use cases

Use Case	isEnoughBalance
Actor	None
Description	카드의 잔액이 음료를 구매할 만큼 충분한지 확인한다.

Use Case	cancelPayment
Actor	None
Description	사용자의 결제를 취소한다.

Use Case	refund
Actor	None
Description	결제 도중 음료의 재고가 사라진 경우 환불한다.

Activity 1006.

Define Business Use Case

Describe use cases

Use Case	prePay
Actor	None
Description	타 DVM에 있는 음료에 대한 선결제를 진행한다.

Use Case	setDVMInfo
Actor	관리자
Description	DVM의 id와 위치정보를 등록한다.

Use Case	getNearestDVM
Actor	None
Description	사용자가 선택한 음료를 보유하고 있는 DVM 중 가장 가까운 것의 위치를 반환한다.

Activity 1006.

Define Business Use Case

Describe use cases

Use Case	createCode
Actor	None
Description	선결제를 위한 인증코드를 생성한다.

Use Case	isVaildCode
Actor	None
Description	인증코드가 유효한지 확인한다.

Use Case	sendReqMsg
Actor	None
Description	다른 DVM에 요청 메시지를 보낸다.

Activity 1006.

Define Business Use Case

Describe use cases

Use Case	sendResMsg
Actor	None
Description	request_message에 대해 응답 메시지를 보낸다.

Use Case	subtractStock
Actor	None
Description	결제가 완료된 음료의 재고를 차감한다.

Use Case	updateStock
Actor	관리자
Description	DVM의 음료 재고를 업데이트 한다.

Activity 1006.

Define Business Use Case

Describe use cases

Use Case	setTotalItemList
Actor	관리자
Description	관리자가 system에서 판매할 전체 음료 20종류의 이름, 음료 코드, 가격을 등록한다.

Use Case	setItemList
Actor	관리자
Description	관리자가 해당 자판기에서 판매할 음료 7종류를 랜덤하게 선택한다.

Use Case	provideItem
Actor	None
Description	사용자가 결제한 음료를 제공해준다.

Activity 1006.

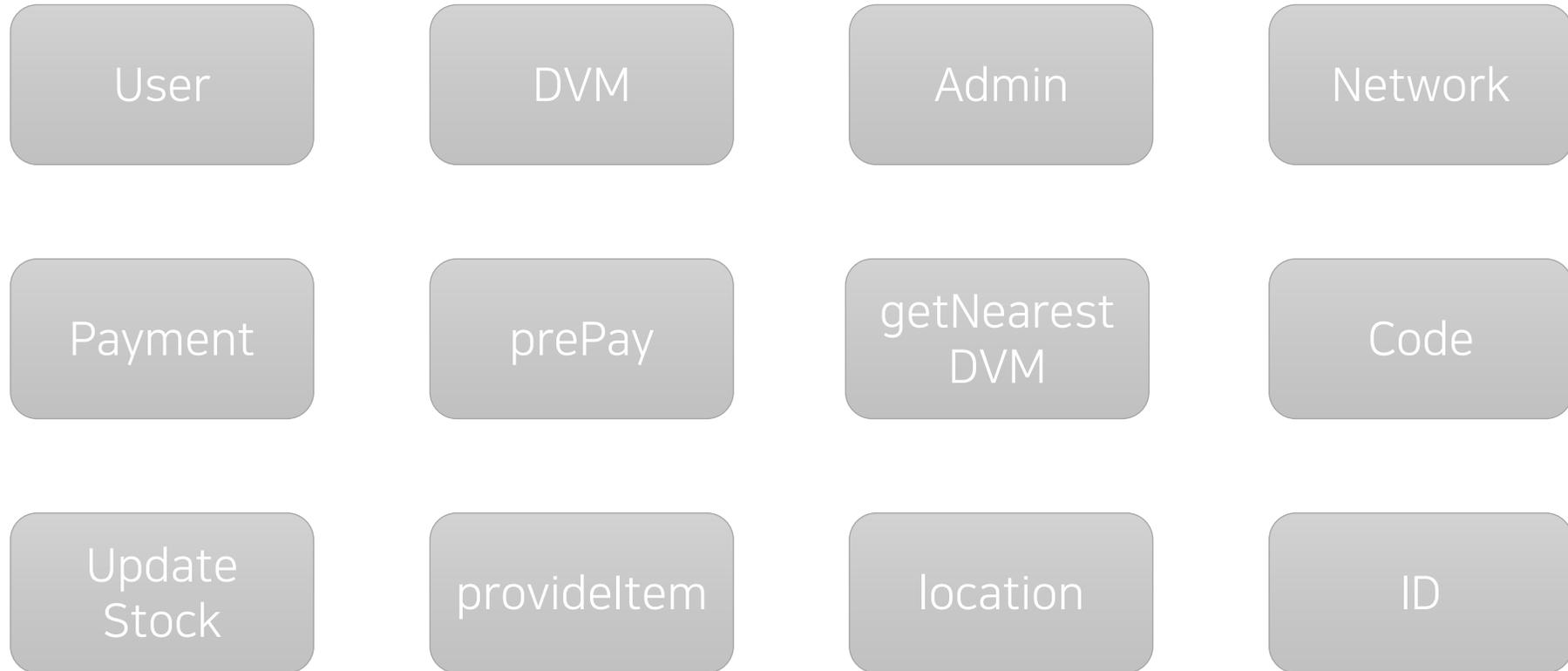
Define Business Use Case

Describe use cases

Use Case	showItemList
Actor	None
Description	판매하는 20종류의 음료를 인터페이스에 출력해준다.

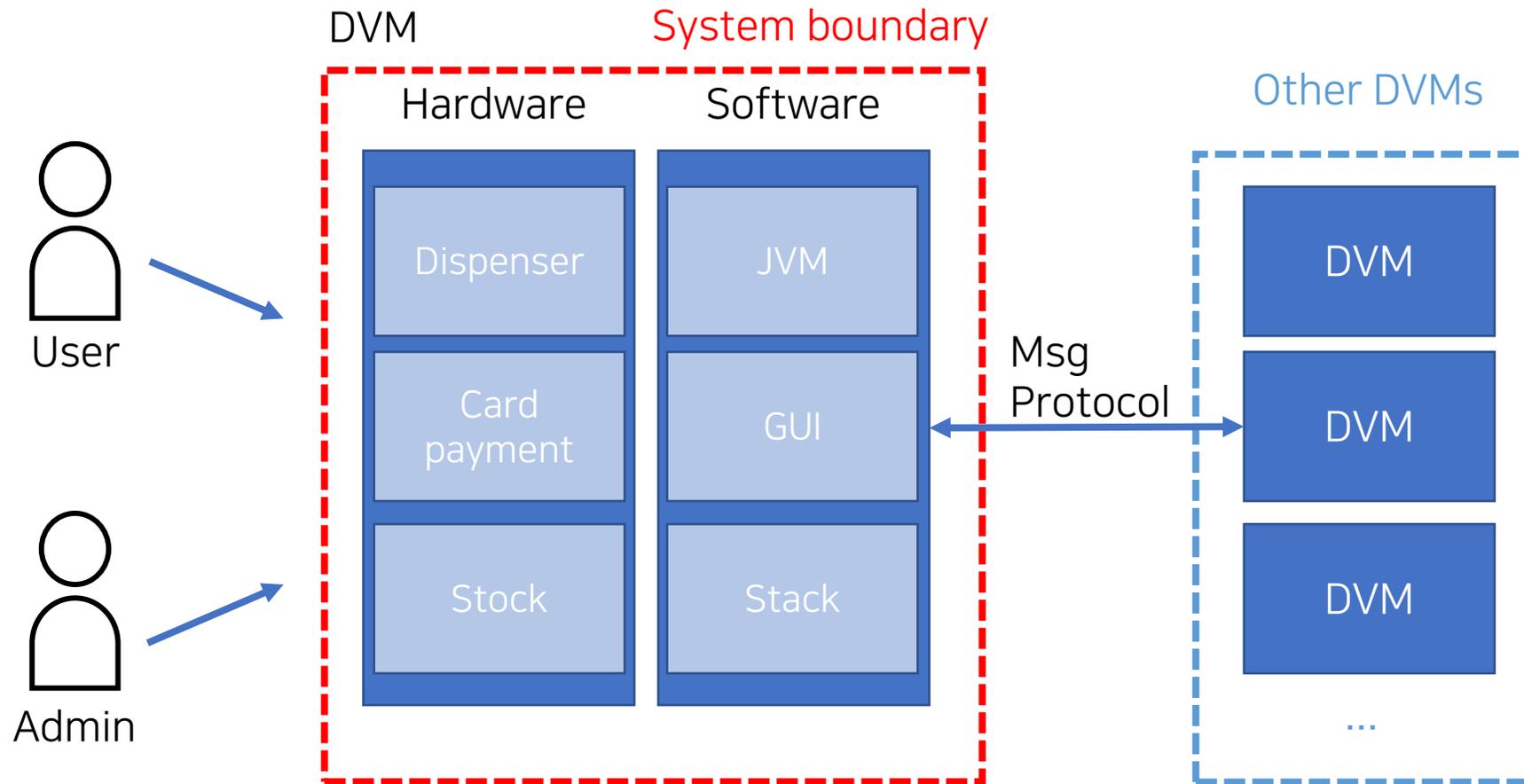
Activity 1007.

Define Business Concept Model



Activity 1008.

Define Draft System Architecture



Activity 1009. Define System Test Case

Ref	Test Use Case	Test Description
R 1.1	selectItem	사용자가 선택한 아이템이 올바르게 선택되는지 확인한다.
R 1.2	isValidCard	사용자가 입력한 카드 번호가 유효하면 True, 아니면 False를 반환하는지 확인한다.
R 1.3	isEnoughBalance	사용자가 입력한 카드의 잔액이 음료 가격보다 크거나 같으면 True, 아니면 False를 반환하는지 확인한다.
R 1.4	buyItem	사용자가 선택한 아이템을 올바르게 결제하는지 확인한다.
R 1.5	cancelPayment	카드번호가 유효하지 않거나, 잔액이 부족하거나, 사용자의 결제를 정상적으로 취소하는지 확인한다.
R 1.6	refund	결제 금액을 정상적으로 환불하는지 확인한다.
R 1.7	prePay	선결제 과정이 정상적으로 진행되는지 확인한다.

Activity 1009. Define System Test Case

Ref	Test Use Case	Test Description
R 2.1	setDVMInfo	관리자가 설정한 DVM 정보가 정상적으로 설정되었는지 확인한다.
R 2.2	getNearestDVM	다른 DVM에서 받은 response message들의 좌표들과 내 DVM의 좌표의 차이를 계산해서 가장 가까운 DVM의 위치를 반환하는지 확인한다. 가장 가까운 DVM이 여러 개인 경우 id의 숫자가 작은 자판기의 위치를 반환하는지 확인한다.
R 3.1	createCode	알파벳 소문자와 숫자를 포함하는 10자리 문자열을 반환하는지 확인한다.
R 3.2	isVaildCode	선결제를 마친 DVM에서 보낸 request message의 인증코드들 중 사용자가 입력한 인증코드가 존재한다면 True를, 아니라면 False를 반환하는지 확인한다.
R 4.1	sendReqMsg	DVM 간에 주고 받는 메시지를 확인하여 정해진 프로토콜 대로 통신하는지 확인한다.
R 4.2	sendResMsg	다른 DVM에서 보낸 request message에 대해 정해진 프로토콜 대로 통신하는지 확인한다.

Activity 1009. Define System Test Case

Ref	Test Use Case	Test Description
R 5.1	subtractStock	결제가 완료된 음료의 재고가 정상적으로 차감되는지 확인한다.
R 5.2	updateStock	음료의 재고 상태를 정상적으로 업데이트 하는지 확인한다.
R 6.1	provideItem	결제 완료된 음료가 정상적으로 제공되는지 확인한다.
R 6.2	setTotalItemList	system에서 판매할 20종류의 음료가 관리자가 등록한 대로 설정되는지 확인한다.
R 6.3	setItemList	각 DVM에서 판매할 7종류의 음료가 정상적으로 랜덤 선택이 되는지 확인한다.
R 7.1	showItemList	판매 중인 20종류의 음료가 정상적으로 인터페이스에 출력되는지 확인한다.

Q&A