

# OOPT Stage 1000 - Plan

: Distributed vending machine

202211252 고승우  
202211300 박연주  
202211349 이채유  
202214202 김민석

# Activity 1001. Define Draft Plan

## Motivation

1. 자판기가 원하는 음료수의 재고가 없을 경우 선택의 폭 제한
2. 원하는 음료수를 판매하고 있는 자판기의 재고 파악이 어려움
3. 원하는 음료수를 판매하고 있는 가장 가까운 자판기의 위치를 알 수 없어 시간 및 동선 낭비

## Project Objectives

새로운 자판기 시스템 소프트웨어는 효율적이고 사용하기 쉬워야한다:

1. 분산 시스템을 통해 자판기 연결 -> 자판기 시스템의 실시간 재고 파악에 용이
2. 다른 자판기 위치 안내 -> 사용자에게 원하는 음료수를 판매하고 있는 가장 가까운 자판기의 위치를 알려줌으로써 효율적인 음료 구매 가능
3. 선결제 시스템 제공 -> 사용자가 음료를 사전에 구입하여 음료가 다른 자판기에서 품절 되는 상황 방지 가능

# Activity 1001. Define Draft Plan

## Functional Requirements

1. 음료 종류 제공
2. 음료 선택
3. 음료 재고 파악
4. 결제
5. 선결제
6. 잔고 부족 시 결제 실패
7. 인증 코드 전송
8. 인증 코드 입력
9. 실시간 음료 재고 반영
10. 자판기 위치 안내

# Activity 1001. Define Draft Plan

## Non-Functional Requirements

1. user가 이해하고 사용하기 쉬운 UI
2. 응답에 대한 반응 시간이 3초 이내
3. 주요 기능이 고장으로 인한 중단 없이 실행
4. 음료수 추가 등록 및 자판기 추가 등록에 용이
5. 사용자의 결제 정보 및 개인 정보 보호
6. msg protocol 준수

# Activity 1001. Define Draft Plan

## Resource Estimation

1. Human resource : 4 member
2. Project duration: about 2 months

## Managerial Issues

1. 짧은 개발 기간
2. 안정적인 네트워크 필요

# Activity 1002. Create Preliminary Investigation Report

## Alternative Solutions

1. 현재 시장에 존재하는 자판기 관리 SW구매했어 기능을 추가 및 개선
2. 외부 개발사에 의뢰

## Project Justification (Business needs)

1. 기존 자판기보다 사용자에게 더 넓은 폭의 음료 선택지 제공 가능 (Effect)
2. 직접 개발함으로써 비용 절감 (Cost)

# Activity 1002. Create Preliminary Investigation Report

## Risk Management

Risk	Probability	Significance	Weight
java 개발 능력 미숙	3	4	12
OOPT 방법론 적용 미숙	4	4	16
개인 일정으로 인한 시간 부족 (Team Commnication)	3	3	9
요구사항 변경	1	4	4

# Activity 1002. Create Preliminary Investigation Report

## Risk Reduction Plan

Risk	
java 개발 능력 미숙	문제 상황에 대한 적극적인 질문과 소통을 통해 문제 해결
OOPT 방법론 적용 미숙	수업 자료 및 추가 자료 검색 및 적극적인 질문
개인 일정으로 인한 시간 부족	일정 조율에 힘쓰고 주기적인 소통
요구사항 변경	초기에 요구사항 명확하게 분석



# Activity 1002. Create Preliminary Investigation Report

## Market Analysis

1. 다른 자판기의 재고 및 위치를 알려주는 자판기 시스템의 부재
2. 해당 소프트웨어를 통해 폭 넓은 음료 선택지 제공 가능

## Managerial Issues

1. 짧은 개발 기간
2. 안정적인 네트워크 필요

# Activity 1003. Define Requirements

## - Functional Requirements

### 1. 고객은 자판기의 음료를 구매할 수 있다.

- 고객은 자판기에서 구매할 음료의 종류와 개수를 선택할 수 있다.
- 한 번에 구매할 수 있는 음료의 개수는 5개 이하이다.
- 고객이 구매하고자 하는 음료가 현재 자판기에서 취급중이며, 재고가 남아있다면 고객은 현재 자판기에서 결제를 진행하고 음료를 제공받는다.
- 음료를 제공하게 되면 그 음료에 대한 재고를 업데이트한다.
- 고객은 현재 자판기가 취급하지 않거나 재고가 없는 음료도 선택할 수 있으며, 이는 선결제로 이어진다.

### 2. 고객은 카드를 이용해 음료를 결제할 수 있다.

- 고객은 카드 번호 입력을 통해 음료를 카드결제할 수 있다.
- 카드사는 고객이 결제를 진행하고자 하는 카드의 잔액 정보를 자판기에 넘겨준다.
- 자판기는 카드 잔액이 결제금액보다 부족하면 결제 진행이 되지 않도록 한다.

# Activity 1003. Define Requirements

## - Functional Requirements

### 3. 고객은 자판기에서 선결제를 할 수 있다.

- 현재 자판기에서 취급하지 않거나 재고가 없는 음료에 한해서 선결제를 진행할 수 있다.
- 다른 자판기와 통신하여 해당 음료의 재고와 위치를 확인하고 선결제 요청을 할 수 있다.
- 고객이 선결제를 요청하는 경우 현재 자판기에서 결제가 진행되고, 음료를 가져갈 자판기의 재고가 업데이트 된다.
- 고객은 음료의 재고가 충분한 다른 자판기 중 가장 가까운 자판기에서 음료를 받아갈 수 있다.
- 선결제가 완료되면 인증코드와 선결제를 마친 자판기의 위치 정보를 안내한다.
- 선결제가 완료된 자판기로 가서 인증코드를 입력하면 음료를 받아갈 수 있다.
- 마찬가지로, 타 자판기에서 현재 자판기에서의 선결제가 완료되었다면, 인증코드를 입력하고 음료를 받아갈 수 있다.

# Activity 1003. Define Requirements

## - Functional Requirements

### 4. 관리자는 자판기에 대한 재고를 관리할 수 있다.

- 관리자 모드로 진입하기 위해 관리자 아이디, 패스워드를 입력할 수 있다.
- 관리자 인증이 확인되면 각 음료에 대한 수량, 가격을 조정할 수 있다.

# Activity 1003. Define Requirements

## - Functional Requirements (Categorized Table)

Ref.	Function	Category
R 1.1	Show Menu	Event
R 1.2	Select Menu	Event
R 1.3	Show Item	Event
R 1.4	Choose Item	Event
R 1.5	Provide Item	Event
R 2.1	Pay	Hidden
R 2.2	Valid Card	Event
R 2.3	Check Stock	Hidden
R 2.4	Update Stock	Hidden

# Activity 1003. Define Requirements

## - Functional Requirements (Categorized Table)

Ref.	Function	Category
R 3.1	Prepay	Event
R 3.2	Check other DVMs	Hidden
R 3.3	Request from others DVMs	Hidden
R 3.3	Provide Authentication Code	Event
R 3.4	Provide Available DVM Location	Event
R 3.5	Check Authentication Code	Event
R 4.1	Admin Login	Event
R 4.2	Admin Logout	Event
R 4.3	Set Item Stock	Event
R 4.4	Set Item Price	Event

# Activity 1003. Define Requirements

## - Non-Functional Requirements

1. Usability (사용성): user가 이해하고 사용하기 쉬운 인터페이스를 통해 소프트웨어를 제공해야한다.
2. Efficiency (효율성): user의 응답에 대한 반응 시간이 3초 이내여야한다.
3. Reliability (신뢰성): 주요 기능이 고장으로 인한 중단 없이 실행되어야한다.
4. Scalability (확장성): 음료수 추가 등록 및 자판기 추가 등록에 용이해야한다.
5. Security (보안성): 사용자의 결제 정보 및 개인 정보가 보호되어야한다.
6. Integrity (무결성): msg protocol을 준수해야한다.

# Activity 1004. Record Terms in Glossary

**DVM:** Distributed Vending Machine로, 분산 시스템을 통해 서로 통신하고 있는 각각의 자판기

**system:** 분산 시스템을 통해 서로 통신하고 있는 자판기 시스템 전체를 의미

**결제 (Pay):** 사용자가 선택한 음료에 대해 금액을 지불하는 행위

**선결제 (Prepay):** 자판기에 음료 재고가 없어 다른 DVM에서 구매해야 할 경우, 상품을 실제로 수령하기 전 결제를 우선 완료하는 행위로 인증 코드 발급을 통해 이루어짐

**제품 예약:** 사용자가 원하는 제품을 선택한 후, 해당 제품이 다른 DVM에 존재할 시 미리 예약하는 것을 의미

**재고관리:** DVM의 제품 재고 상태를 관리한다.

**인증 코드 (Authentication Code):** 선결제를 완료한 사용자가 제품을 수령하고자 할 때 필요한 코드로, 다른 DVM으로 가서 해당 코드를 입력하면 음료를 수령할 수 있음

**사용자 (User):** DVM에서 음료를 구매하고자 하는 사람

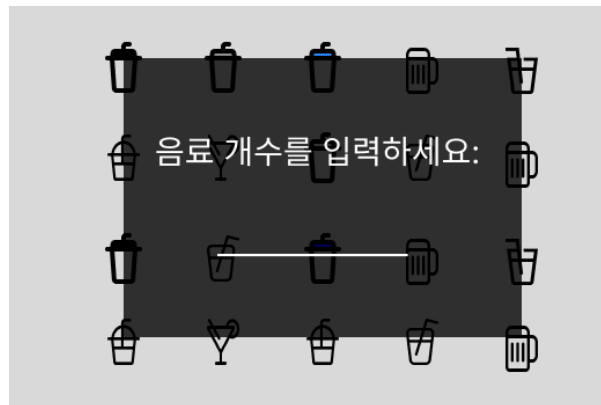
**관리자 (Administrator):** DVM을 관리하는 사람

**관리자 모드:** 관리자가 DVM에서 판매할 음료의 재고 및 가격을 설정하는 모드

**카드사 (Card Company):** 사용자의 결제 정보를 처리하는 외부 기관으로 DVM 시스템은 카드사를 통해 결제를 안전하게 처리

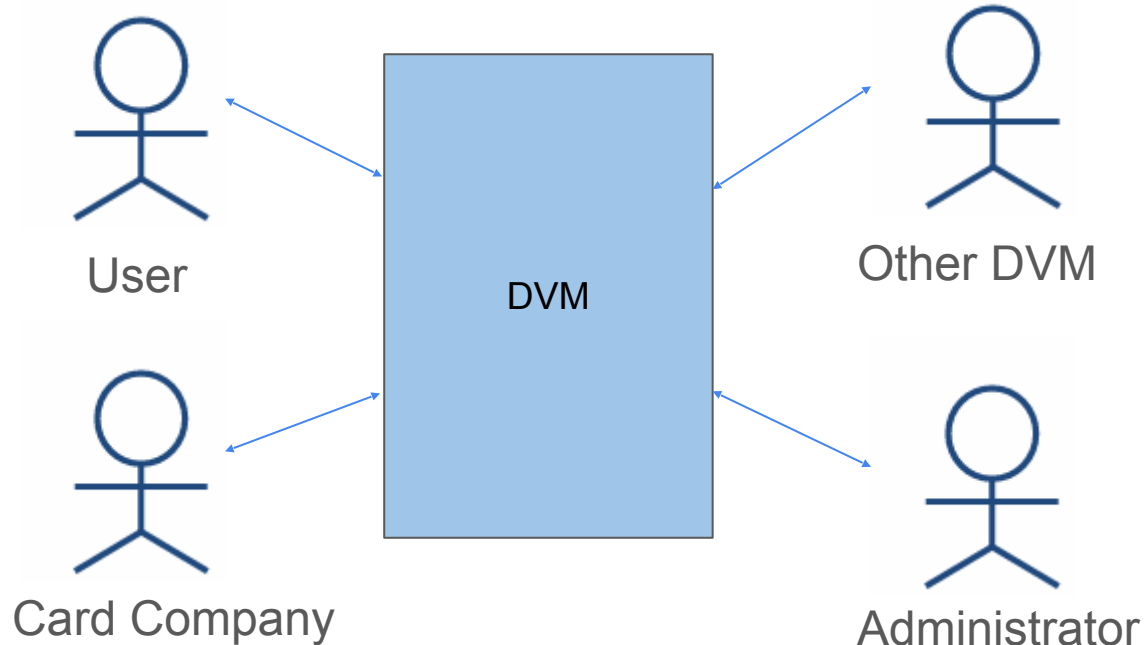


# Activity 1005. Implement Prototype



# Activity 1006. Define Business Use Case

- Define system boundary



# Activity 1006. Define Business Use Case

- Identify and describe actors

- Primary Actor

- : User - 음료를 구매하기 위해 시스템과 상호작용하는 actor

- : Administrator: 자판기 음료의 재고를 관리하기 위해 시스템과 상호작용하는 actor

- Supporting Actor

- : Other DVM - 선결제를 위해 음료 재고와 위치 전달을 위해 시스템과 상호작용하는 actor

- : Card Company: 카드 결제에서 잔액 정보 전달을 위해 시스템과 상호작용하는 actor

# Activity 1006. Define Business Use Case

- Identify use cases

1. Use-cases by actor-based (User)

- Select Menu, Show Item, Choose Item, Provide Item, Valid Card, Check Authentication Code

2. Use-cases by actor-based (Card Company)

- Valid Card, Pay, Prepay

3. Use-cases by actor-based (Other DVM)

- Check other DVMs, Provide Authentication Code, Provide Available DVM Location, Request from other DVMs

4. Use-cases by actor-based (Administrator)

- Admin Login, Admin Logout, Set Item Stock, Set Item Price

5. Use-cases by event based

- Show Menu, Update Stock

# Activity 1006. Define Business Use Case

- Use cases

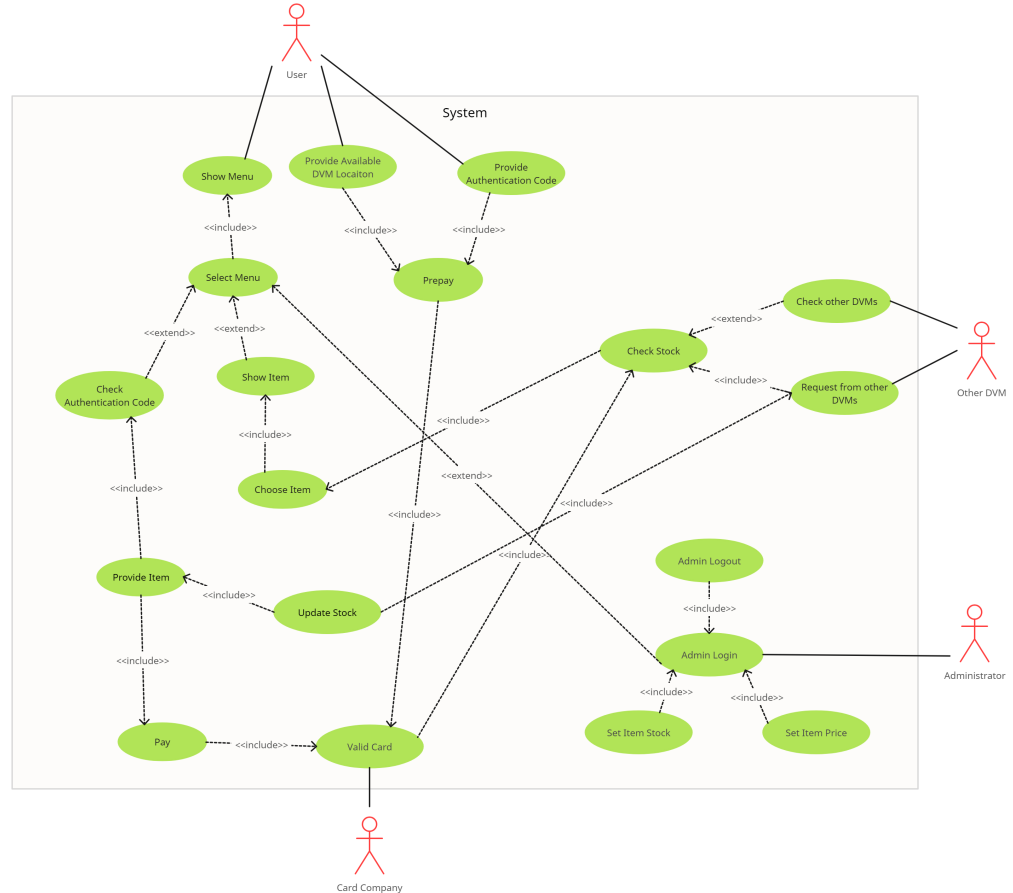
Ref.	Function	Use Case Number & Names	Category
R 1.1	Show Menu	1. Show Menu	Primary
R 1.2	Select Menu	2. Select Menu	Primary
R 1.3	Show Item	3. Show Item	Primary
R 1.4	Choose Item	4. Choose Item	Primary
R 1.5	Provide Item	5. Provide Item	Primary
R 2.1	Pay	6. Pay	Primary
R 2.2	Valid Card	7. Valid Card	Primary
R 2.3	Check Stock	8. Check Stock	Primary
R 2.4	Update Stock	9. Update Stock	Primary

# Activity 1006. Define Business Use Case

Ref.	Function	Use Case Number & Names	Category
R 3.1	Prepay	10. Prepay	Primary
R 3.2	Check other DVMs	11. Check other DVMs	Primary
R 3.3	Request from other DVMs	12. Request from other DVMs	Secondary
R 3.4	Provide Authentication Code	13. Provide Authentication Code	Primary
R 3.5	Provide Available DVM Location	14. Provide Available DVM Location	Primary
R 3.6	Check Authentication Code	15. Check Authentication Code	Primary
R 4.1	Admin Login	16. Admin Login	Secondary
R 4.2	Admin Logout	17. Admin Logout	Secondary
R 4.3	Set Item Stock	18. Set Item Stock	Primary
R 4.4	Set Item Price	19. Set Item Price	Primary

# Activity 1006. Define Business Use Case

- Draw a use case diagram



# Activity 1006. Define Business Use Case

- Describe use cases

Use case	1. Show Menu
Actors	None
Description	음료 선택, 인증코드 입력, 관리자 모드 중 하나를 선택할 수 있는 메뉴를 출력한다.

Use case	2. Select Menu
Actors	User
Description	시스템이 출력한 메뉴인 음료 선택, 인증코드 입력, 관리자 모드 중 하나를 선택한다.

Use case	3. Show Item
Actors	None
Description	DVM에서 판매하는 음료들의 목록을 출력한다.



# Activity 1006. Define Business Use Case

- Describe use cases

Use case	4. Choose Item
Actors	User
Description	음료 목록 중 사용자가 원하는 음료의 종류와 그 음료의 개수를 선택한다.

Use case	5. Provide Item
Actors	None
Description	사용자가 선택한 음료를 개수에 맞게 사용자에게 제공한다.

# Activity 1006. Define Business Use Case

- Describe use cases

Use case	6. Pay
Actors	None
Description	사용자에게 입력받은 카드 정보를 바탕으로 결제를 진행한다.

Use case	7. Valid Card
Actors	User
Description	사용자에게 입력받은 카드 정보가 유효한지 확인하고, 입력받은 카드에 얼마 만큼의 잔액이 남았는지 계산한다. 이 때, 잔액이 충분하면 결제에 성공하고 잔액이 충분하지 않으면 결제에 실패한다.

# Activity 1006. Define Business Use Case

- Describe use cases

Use case	8. Check Stock
Actors	None
Description	사용자가 선택한 음료 혹은 다른 DVM에서 요청한 음료의 재고가 있는지 파악한다.

Use case	9. Update Stock
Actors	None
Description	사용자가 결제에 성공해 음료를 제공하거나 다른 DVM에서 선결제해 system이 음료를 제공해야 하는 경우 음료의 재고를 업데이트한다.

Use case	10. Prepay
Actors	None
Description	사용자가 선택한 음료가 재고가 DVM에 없거나 판매하는 종류가 아닐 경우 선결제를 진행한다.

# Activity 1006. Define Business Use Case

- Describe use cases

Use case	11. Check other DVMs
Actors	None
Description	사용자가 선택한 음료가 DVM에 없을 때, 사용자에게 음료를 제공할 수 있는지 확인하기 위해 다른 DVM과 통신한다.
Use case	12. Request from Other DVMs
Actors	None
Description	다른 DVM에서 온 요청에 대해 응답한다. 재고 확인 요청이 오면 요청받은 재고 정보와 위치를 반환하고, 선결제 요청이 왔을 때 선결제가 가능하다면, 인증코드를 받고 음료를 판매된 것으로 처리한다.
Use case	13. Provide Authentication Code
Actors	None
Description	다른 DVM에서 음료를 받을 수 있게 사용자에게 인증 코드를 발급하고 제공한다.

# Activity 1006. Define Business Use Case

- Describe use cases

Use case	14. Provide Available DVM Location
Actors	None
Description	사용자가 선택한 음료가 DVM에 없을 때, 그 음료를 제공해줄 수 있는 다른 DVM의 위치를 찾고 위치를 사용자에게 제공한다.

Use case	15. Check Authentication Code
Actors	User
Description	사용자가 입력한 인증코드를 다른 DVM에게서 전달받은 인증코드와 같은 지 검사한다.

# Activity 1006. Define Business Use Case

- Describe use cases

Use case	16. Admin Login
Actors	Administrator
Description	음료에 대한 재고 및 가격을 조정할 수 있는 관리자 모드로 로그인한다.

Use case	17. Admin Logout
Actors	Administrator
Description	관리자 모드에서 로그아웃한다.

# Activity 1006. Define Business Use Case

- Describe use cases

Use case	18. Set Item Stock
Actors	Administrator
Description	관리자모드일 때, 관리자가 음료에 대한 재고를 조정할 수 있다.

Use case	19. Set Item Price
Actors	Administrator
Description	관리자모드일 때, 관리자가 음료에 대한 가격을 조정할 수 있다.

# Activity 1007. Define Business Concept Model

사용자

음료

관리자

결제

다른 DVM

인증

네트워크

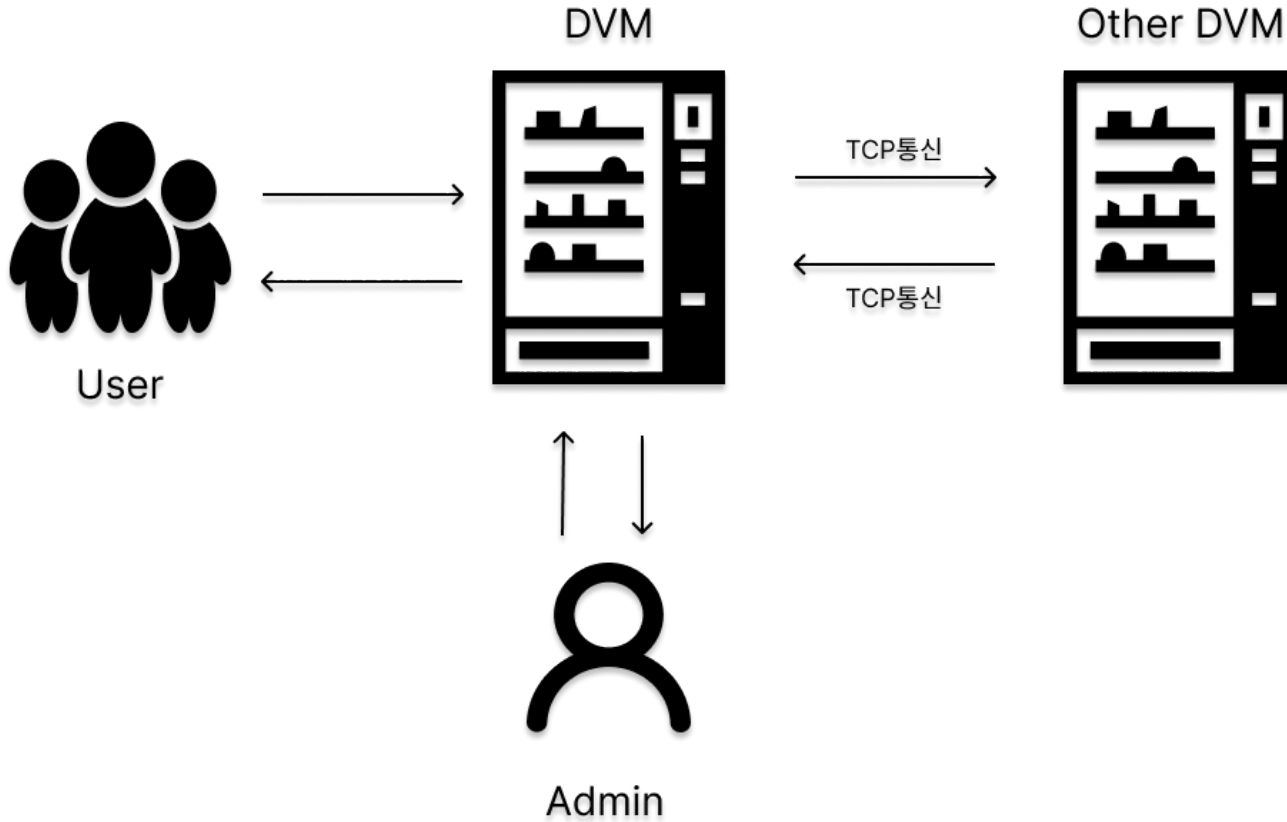
재고

선결제

카드



# Activity 1008. Define Draft System Architecture



# Activity 1009. Define System Test Case

Test Number	Test 항목	Description	use Case	System Function
1	PrintMenu	메뉴 20가지가 잘 출력되는지 확인	2	R 1.2
2	Choose Item	선택된 음료 개수와 종류를 확인	4	R 1.4
3	Provide item	음료 개수와 종류가 알맞게 제공됐는지 확인	5	R 1.5
4	Payment	결제 진행 점검	6	R 2.1
5	Valid Card	카드에 잔액이 음료가격보다 큰지 확인	7	R 2.2

# Activity 1009. Define System Test Case

Test Number	Test 항목	Description	use Case	System Function
6	Update Stock	결제 후 음료 재고 업데이트 됐는지 확인	9	R 2.4
7	Update Stock	다른 자판기로부터 선결제 확인 시 재고 차감 확인	9	R 2.4
8	PrePayment	선결제 요청 점검	10	R 3.1
9	Check other DVMS	재고가 없을 시 다른 DVM과 통신이 되는지 확인	11	R 3.2

# Activity 1009. Define System Test Case

Test Number	Test 항목	Description	use Case	System Function
10	Request from other DVMs	다른 자판기에 구매 정보 및 인증코드 송신 점검	13	R 3.3
11	Provide Authentication Code	선결제시 인증코드가 발급되는지 확인	13	R 3.4
12	Provide Available DVM Location	선결제된 DVM 위치를 알맞게 주었는지 확인	14	R 3.5
13	Check Authentication Code	입력한 인증코드가 DVM이 발급한 인증코드와 같은지 확인	15	R 3.6

# Activity 1009. Define System Test Case

Test Number	Test 항목	Description	use Case	System Function
16	Admin Login	관리자모드로 접속 되는지 확인	16	R 4.1
17	Admin Logout	관리자모드 로그아웃 기능 확인	17	R 4.2
18	Set Item Stock	관리자모드, 음료 재고 조정 확인	18	R 4.3
19	Set Item Price	관리자모드, 음료 가격 조정 확인	19	R 4.4

# Activity 1010. Refine Plan

- 프로젝트 범위

- DVM(Distributed Vending Machine)은 해당 자판기 재고 소진 시 가까운 자판기로 안내하는 시스템을 자동화 합니다. 선결제, 인증코드 생성/확인을 포함합니다.

- 프로젝트 목표

- 자동화된 DVM시스템을 개발하여 다음과 같은 작업을 제공합니다
  - 선결제 요청 및 인증코드 발급, 가까운 DVM위치 안내
- DVM은 가장 가까운 자판기의 위치 안내가 효율적이어야 합니다

# Activity 1010. Refine Plan

- **성능 요구사항**

- 다른 DVM와 통신 시 5초 이내 이용 가능한 자판기를 확인해야합니다

- 결제시 5초 이내 반영되어야 합니다

- 선결제시 5초 이내 인증코드 발급 및 전송이 이루어져야 합니다.

- **운영환경**

- Microsoft windows

- **사용자 인터페이스**

- 음료 메뉴 기반 접근 방식

- Window기반 버전으로 업그레이드 설계

# Activity 1010. Refine Plan

- 기타 요구사항

- 사용자는 관리자 모드 접근을 제한해야 합니다
- 재고의 내용은 신뢰할 수 있게 유지되어야 합니다

- 소프트웨어

- 운영체제: windows
- 프로그래밍 언어: java
- 케이스 도구: intellij IDEA

- 자원

- 인력: 4명
- 기간: 2달



# Activity 1010. Refine Plan

- **Configuration Management**

- 버전 관리 시스템(Version Control System, VCS) → Git사용예정
- Directory Management

- **Quality Assurance Plan(품질 보증 계획)**

- 유닛 테스트(Unit Testing Plan) :각 구성 요소가 기대한 대로 작동하는지 확인하기 위한 테스트 계획 수립
- 통합 테스트(Integration Testing Plan) :개별 구성 요소들을 통합하여 전체 시스템이 예상대로 작동하는지 확인하는 테스트 계획 수립
- 품질 측정 지표(Quality Metrics) :자판기의 품질을 측정하고 평가하기 위한 지표를 정의