

**OOPT 1000: Planning**

**Distributed Vending Machine**

**TEAM 1**

팀장 정연수

팀원 김민환 이승현 조벽정 황유란

## Index

1001. Define Draft Plan	3 ... 3
1002. Create Preliminary Investigation Report	4 ... 5
1003. Define Requirements	6 ... 6
1004. Record Terms in Glossary	7 ... 7
1005. Implement Prototype	8 ... 8
1006. Define Business Use Case	9 ... 13
1007. Define Business Concept Model	14 ... 14
1008. Define Draft System Architecture	15 ... 16
1009. Refine Plan	17 ... 20

## 1001. Define Draft Plan

- Motivation
  - 팀원 중 한 명이 자판기를 이용할 때 먹고 싶은 음료를 찾기 위해서 학교 여러 군데를 돌아다니는 불편함이 있었다.
  - 원하는 카페를 찾아가듯 내가 먹고싶은 음료가 있는 자판기의 위치를 알 수 있으면 좋겠다는 욕구가 있었다.
  - 자판기의 단가를 최대한 높이지 않기 위해 중앙 관리 서버를 두지 않고 각 자판기끼리 연결되어 통신하는 분산 자판기 시스템을 만든다.
- Project Objectives
  - 사용자가 원하는 음료를 제공한다.
  - 사용자가 원하는 음료를 가지고 있는 자판기의 위치를 제공한다.
  - Stand alone 하게 작동하던 자판기를 네트워크로 연결함으로써 많은 이익을 얻는다.
- Functional Requirements
  - Payment : 결제
  - Inform Location : 자판기 위치 정보 제공
  - Advance Payment Verification Code : 선구매 인증번호 처리
  - Network Message : 네트워크 Message 처리
  - Stock Update : 자판기 재고 최신화
  - Vending Item : 음료 제공
- Non Functional Requirements
  - 시중에 존재하는 fresh store 자판기를 벤치마크하여 UI를 제공한다.



- Java 사용(version : JDK 1.8)
  - 네트워크 통신 중에 투명성과 무결성이 제공된다.
  - 단독망 사용.
- Resource Estimation
    - Human Resource : 5명
    - Human Efforts(Man -Months): 12 - 15 M/M.

- Project Duration : 2개월
- Cost : 15,000 Won \* 10hour = 150,000 Won per Person in 1 week
- Total Cost : 150,000 Won \* 10 week \* 5 people = 7,500,000 Won
- Other Information
  - Future Version
    - 빅데이터 기술을 적용시켜 고객 맞춤 음료 제공
    - 학교 Network와 연결

### 1002. Create Preliminary Investigation Report

- Alternative Solutions
  - 편의점
  - 쿠팡/SSG/배민 등 배달
  - 카페
- Project Justification (Business Demand)
  - 최저시급 증가로 인한 인건비 부담
  - 일회용품 사용량을 줄일수 있어서 환경 친화적
  - 공간이 부족한 학교에 적합
- Risk Management

Risk	Probability	Significance	Weight
코로나로 인한 대면의 어려움	5	3	15
OOPT에 대한 낮은 이해도	4	5	20
개발 능력 부족	3	4	12
팀워크 부족 및 소통의 부재	4	5	20
다른 과목에 대한 부담감	2	3	6
팀원의 휴학	1	5	5

- Risk Reduction Plan

Risk	Reduction Plan

코로나로 인한 대면의 어려움	Zoom 또는 Google Meeting 을 통한 온라인 회의 진행
OOPT에 대한 낮은 이해도	실습간에 개념 복기, 수업자료 복습
개발 능력 부족	구글링을 통한 문제해결, 잘 짜여진 코드 리뷰
팀워크 부족 및 소통의 부재	카카오톡으로 실시간 소통, 솔직하게 의견을 낼 수 있는 분위기 형성
다른 과목에 대한 부담감	녹화 강의 제 시간에 듣고 과제 미리미리 시작하기
팀원의 휴학	진심으로 설득하기

- Market Analysis
  - 하나의 자판기에 판매하고자 하는 음료의 종류가 제한이 있다.
  - 다른 대학교에도 상용화 할 수 있다.
- Other Managerial Issues
  - 이 Project는 6월 11일 전에는 완수해야 한다.

### 1003. Define Requirements

- Functional Requirements

Ref	Function	Category
1.1	Set Up All	Hidden
1.2	Select Mode (모드 설정)	Evident
1.3	Manage Stock (재고 관리)	Evident
2.1	Show Item (음료 정보 제공)	Hidden
2.2	Select Item (음료 선택)	Evident
2.3	Check Stock Count (재고 수량)	Hidden
2.4	Update Stock (재고 최신화)	Hidden
2.5	Purchase (현장구매 요청 처리)	Evident
2.6	Advance Purchase (선구매 요청 처리)	Evident
2.7	Check Payment (결제 확인)	Hidden
2.8	Select Advance Payment (선구매 여부 선택)	Evident
2.9	Location Inform (위치 정보 제공)	Hidden
2.10	Create Verification Code (인증번호 생성)	Hidden
2.11	Show Verification Code (인증번호 표시)	Hidden
2.12	Item Out (상품 지급)	Hidden
3.1	Read Verification Code (인증번호 읽기)	Evident
3.2	Check Verification Code (인증번호 확인)	Hidden
3.3	Reset Verification Code (인증번호 초기화)	Hidden
4.1	Message Request ( 다른 자판기에 메시지 요청 )	Hidden
4.2	Message Response ( 다른 자판기에 메시지 응답 )	Hidden

- Performance Requirements

- 자판기간 통신 응답 시간이 1초 이내로 수행되어야 한다.
- 데이터를 확인하는 시간이 1초 이내로 수행되어야 한다.

- Operating Environment

- JVM and Windows 10

- Interface Requirements

- 직관적으로 사용할 수 있어야 한다.

- Other Requirements

- 다른 자판기와 통신되지 않더라도 혼자 동작될 수 있어야 한다.

**1004. Record Terms in Glossary**

Term	Description	Remarks
Item	자판기에 판매되는 음료	
Purchase	결제 유무	
User	분산 자판기를 사용하는 사용자	
Message	네트워크 메시지	
Msg	Message의 약어	
Verification Code	선구매 인증번호	
Count	item의 수량	
Mode	사용자에게 제공하는 구매 유형.	
VM	Vending Machine	
ID	자판기를 구별하기 위한 고유의 아이디	
Broadcast	자신을 제외한 모든 자판기에게 message를 요청하는 과정	

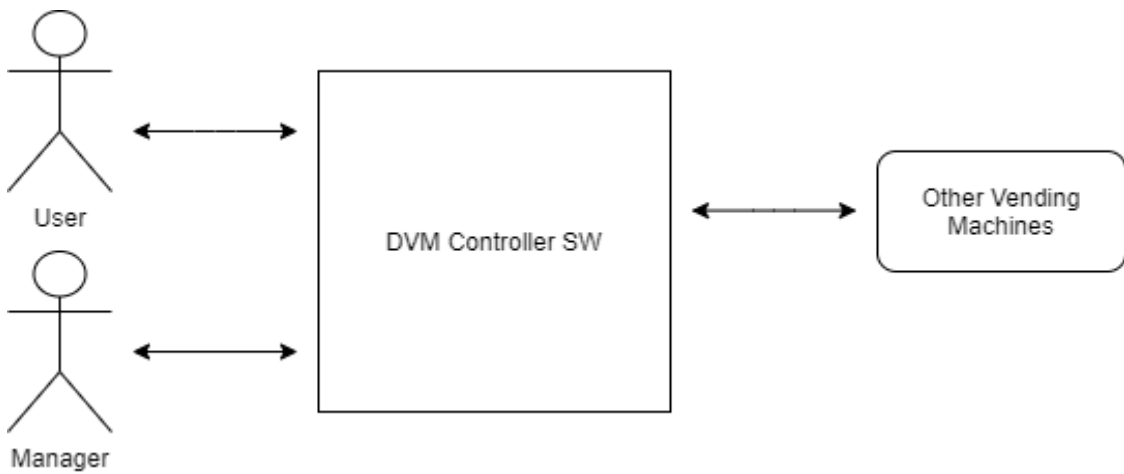
### 1005. Implement Prototype

Select & Use	Check & Inform	Msg	Verification
select mode	advance payment	msg request	create code
select item	payment	msg response	read code
show item	stock count		reset code
advance payment	update stock		
purchase	inform location		
	verification code		



**1006. Define Business Use Case**

- Define system boundary



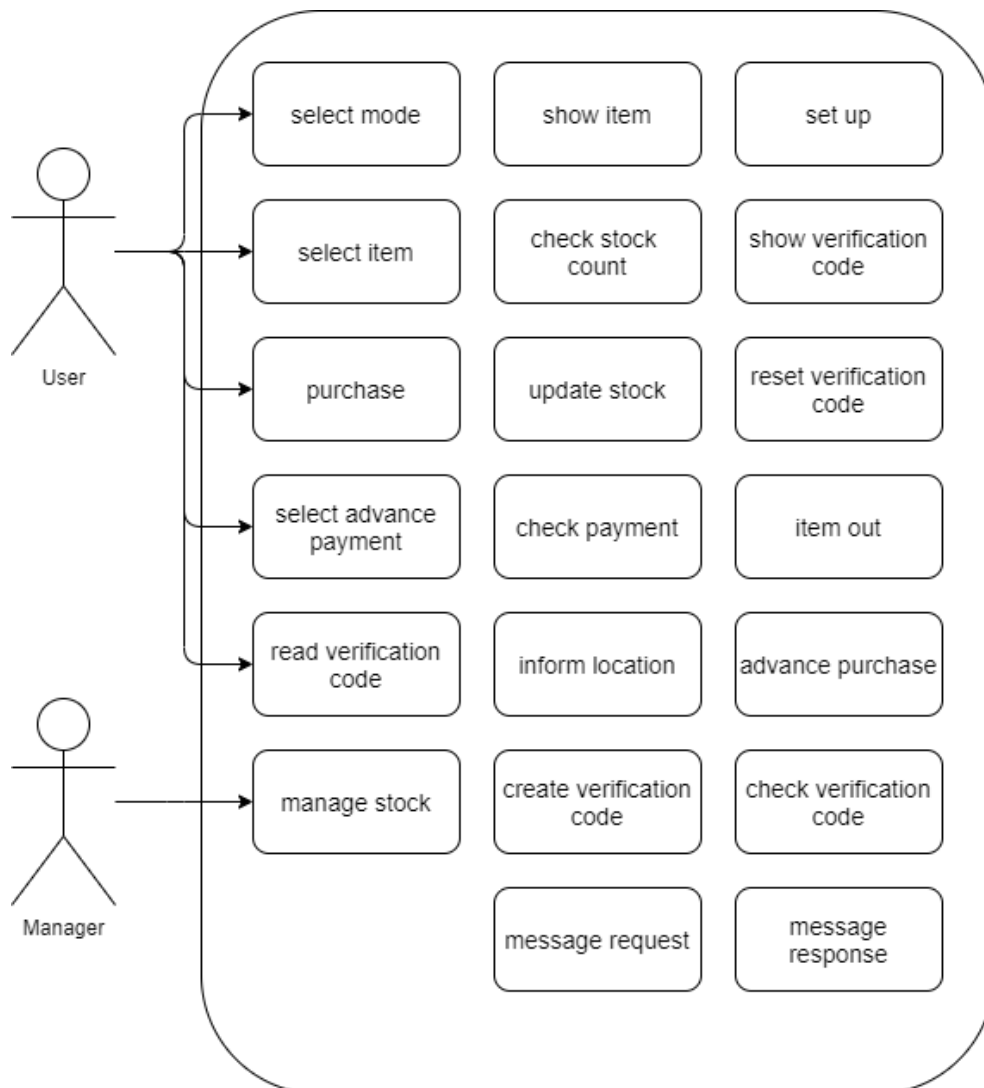
- Identify and describe actors
  - User : 음료를 구매하기 위해 시스템과 상호작용하는 actor
  - Manager : 음료 재고를 보충하는 actor
  - Other Vending Machine : 동일한 네트워크에 연결된 현재 자판기와 통신하는 actor
- Identify use cases
  - Use-cases by actor-based
    - Select Mode, Select Item, Purchase, Select advance payment, Read Verification Code
  - Use-cases by evident-based
    - Set Up, Show Item, Check Stock Count, Update stock, Check Payment, Inform Location, Create code, Show Code, Item Out, Check Verification Code, Reset verification code, Message Request, Message Response
- Allocate system functions into related use cases

Ref	Function	Use Case Number & Name	Category
1.1	Set up all	1. Set up	
1.2	Select mode	2. Select mode	
1.3	Manage stock	3. Manage stock	
2.1	Show item	4. Show item	
2.2	Select item	5. Select item	
2.3	Check stock count	6. Check stock count	
2.4	Update stock	7. Update stock	
2.5	Purchase	8. Purchase	
2.6	Advance purchase	9. Advance purchase	

2.7	Check payment	10. Check payment	
2.8	Select advance payment	11. Select advance payment	
2.9	Location inform	12. Inform location	
2.10	Create verification code	13. Create verification code	
2.11	Show verification code	14. Show verification code	
2.12	Item out	15. Item out	
3.1	Read verification code	16. Read verification code	
3.2	Check verification code	17. Check verification code	
3.3	Reset verification code	18. Reset verification code	
4.1	Message request	19. Message request	
4.2	Message response	20. Message response	

-아래 그림 수정

- Draw a use case diagram



- Describe use cases

Use Case	1. Set Up
Actor	None
Description	- 시스템이 부팅될 때 실행되는 use case. - 시스템 부팅은 관리자가 재고를 보충하거나, 전원에 연결되어 최초 실행될 때를 의미한다.

Use Case	2. Select Mode
Actor	User
Description	- 사용자가 자판기에서 음료를 구매하기 위해 구매 모드를 선택하는 use case.

Use Case	3. Manage Stock
Actor	Manager
Description	- 관리자가 자판기의 재고를 최신화하고 입력한다.

Use Case	4. Show Item
Actor	None
Description	- 사용자에게 모든 음료를 보여준다.

Use Case	5. Select Item
Actor	User
Description	- 사용자가 자판기에서 음료를 선택했을 때 실행되는 use case.

Use Case	6. Check Stock Count
Actor	None
Description	- 사용자가 자판기에서 선택할 음료를 판매하기 위해 재고의 유무 및 수량을 확인하는 use case.

Use Case	7. Update Stock
Actor	None
Description	- 재고의 증감을 하기 위한 use case. - 사용자가 자판기에 음료를 구매하면 해당 음료의 재고를 감소시키거나 관리자가 음료의 재고를 보충하면 해당 음료 재고의 수를 증가시킨다.

Use Case	8. Purchase
Actor	User
Description	- 사용자가 고른 음료를 결제하는 use case.

Use Case	9. Advance Purchase
Actor	User
Description	- 사용자가 고른 음료를 선구매하는 use case.

Use Case	10. Check Payment
Actor	None
Description	- 사용자가 선택한 음료에 대한 결제를 확인하고 음료를 제공하는 use case.

Use Case	11. Select Advance Payment
Actor	User
Description	- 사용자가 선구매 선택하는 use case.

Use Case	12. Inform Location
Actor	None
Description	- 다른 자판기로부터 응답 msg가 도착한 후 실행되는 use case. - 사용자에게 선택한 음료를 판매하고 있는 모든 자판기의 위치를 알려준다.

Use Case	13. Create Verification Code
Actor	None
Description	- 사용자가 선구매를 선택한 경우 실행되는 use case.

	- 6자리 코드를 생성한다..
--	------------------

Use Case	14. Show Verification Code
Actor	None
Description	- 생성한 인증코드를 사용자에게 보여주는 use case.

Use Case	15. Item Out
Actor	None
Description	- 결제 완료된 음료를 제공하는 use case.

Use Case	16. Read Verification Code
Actor	User
Description	- 사용자가 선구매 모드를 통해 발급받은 코드를 자판기에 입력하는 use case.

Use Case	17. Check Verification Code
Actor	None
Description	- 사용자가 입력한 선구매 코드를 검사하는 use case.

Use Case	18. Reset Verification Code
Actor	None
Description	- 입력받은 인증코드의 사용이 끝나면 해당 코드를 삭제하는 use case.

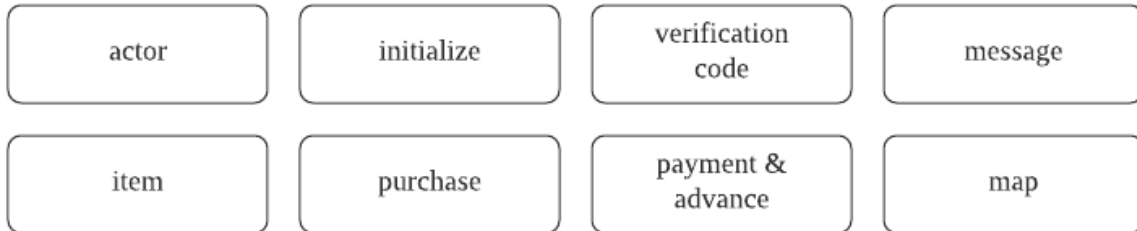
Use Case	19. Message Request
Actor	None Or Other Vending Machine
Description	- 특정 정보에 대한 요청 메시지를 네트워크를 통해 다른 VM에게 보낸다.

Use Case	20. Message Response
Actor	None Or Other Vending Machine

Description	- 받은 요청 메시지에 대한 답을 네트워크를 통해 요청 메시지를 보낸 VM에게 보낸다.
-------------	--

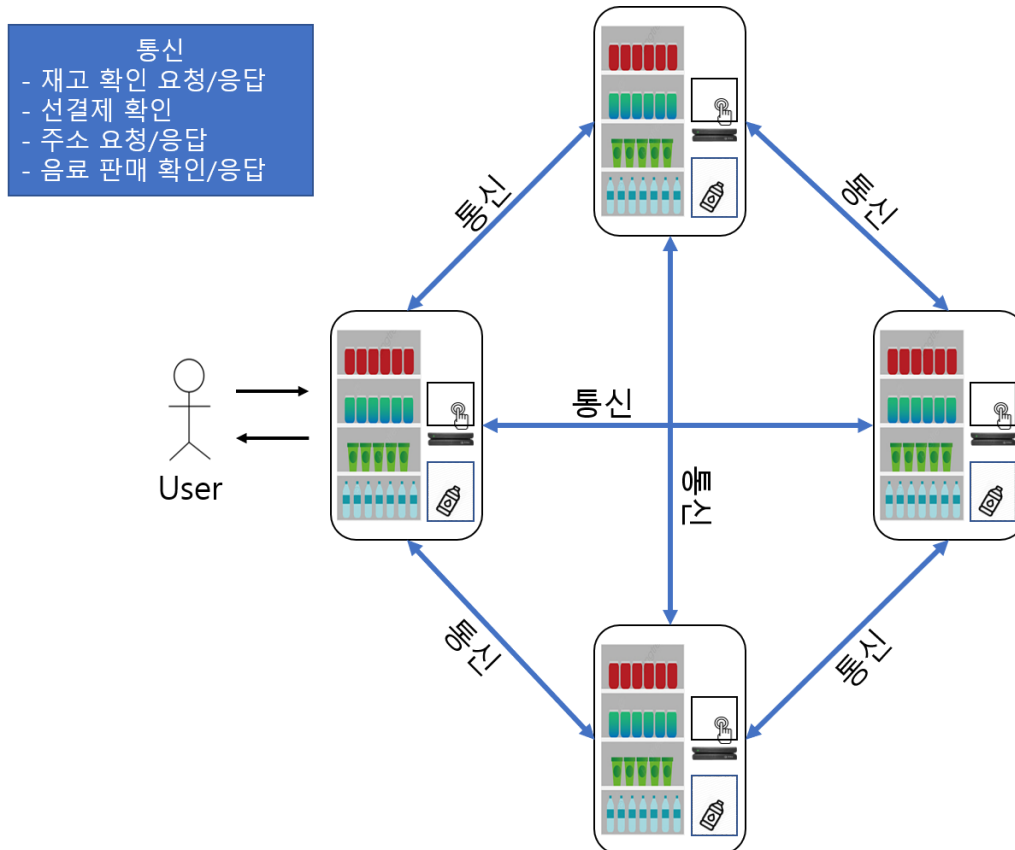
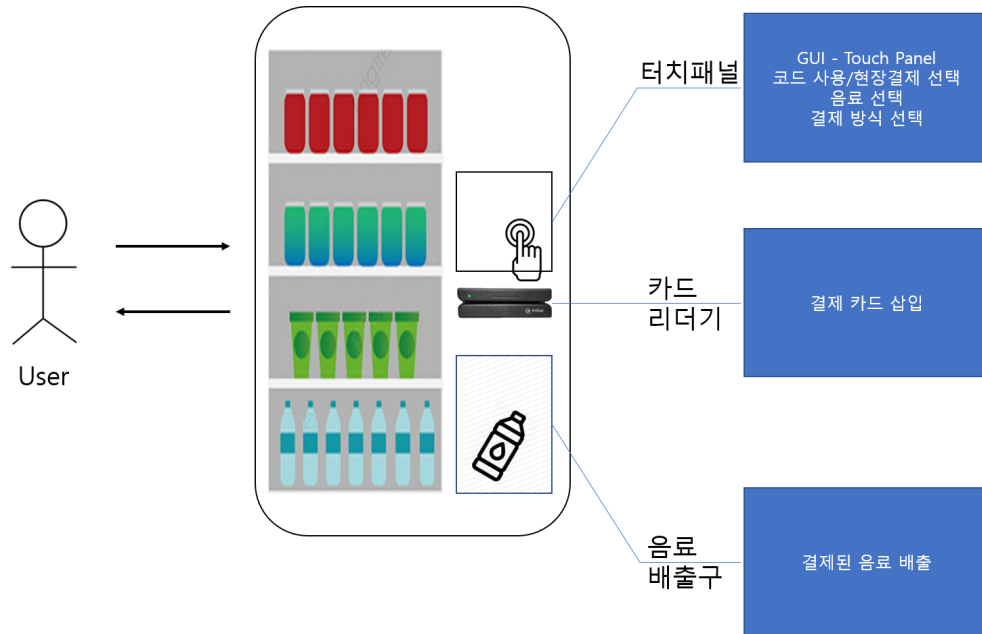
### 1007. Define Business Concept Model

- Identify 'Concept' in the target domain ( Business Modeling )

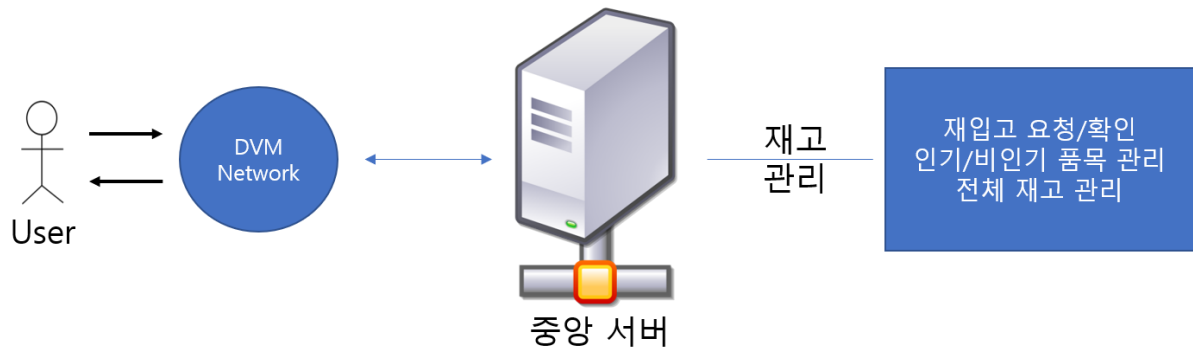


### 1008. Define Draft System Architecture

- Initial version



- Next version





**1009. Refine Plan**

- Project scope
  - 분산 자판기 시스템의 자판기 Controller Software를 개발한다.
- Project objectives
  - 사용자가 원하는 음료를 제공한다.
  - 사용자가 원하는 음료를 가지고 있는 자판기의 위치를 제공한다.
  - Stand alone 하게 작동하던 자판기를 네트워크로 연결함으로써 많은 이익을 얻는다.
- FR

Functional Requirements Test Case			
No.	Function	Test Case	
		Positive	Negative
1	Set up all	1. 전원 공급: 네트워크 및 재고 초기화	1. 전기가 방전: 시스템이 강제 종료됨
2	Select Mode	1. 선구매 코드 입력 모드 선택 : 선구매 모드 진입된다. 2. 현장구매 모드 선택: 현장 구매 모드 진입된다. 3. 재고 관리 선택: 재고 관리 모드 진입된다.	1. 세 모드 동시에 선택: 현장구매 모드 선택 진입된다. 2. 카드 결제: 아무 반응이 없다
3	Manage Stock	1. 기본: 업데이트한 재고를 입력한다.	1. 아무반응 없다.
4	Show Item	1. 기본: 모든 음료가 보여진다. 2. 취소를 누를 경우: Select Mode로 돌아간다	1. 전체 자판기에서 판매하는 모든 음료가 보여지지 않는다

5	Select Item	1. 기본: 선택한 음료의 정보를 바탕으로 다음단계를 진행한다 2. 취소를 누를 경우: Select Mode로 돌아간다.	1. 카드 결제: 아무 반응이 없다.
6	Check Stock Count	1. 기본: 실제 재고와 일치하는 값으로 저장해 다음단계를 진행한다.	1. 현재 자판기에 재고가 없지만, 다른 자판기에 재고가 있는 경우 : : 다른 자판기의 재고 수를 요청한다.
7	Update Stock	1. 기본: 재고를 최신화 한다	1. 기본: 시스템이 재시작 된다
8	Purchase	1. 기본: 입력받은 카드 정보로 결제를 진행한다. 2. 취소를 누를 경우: Select Mode로 돌아간다.	1. 카드가 아닌 다른 것이 리더기에 삽입되는 경우: 오류 메시지를 출력한다.
9	Check Payment	1. 기본: 결제 성공 여부를 반환한다.	1. 기본: 결제 실패를 반환한다.
10	Select Advance Payment	1. 기본: 선구매를 진행한다.	1. 기본: 두 손으로 동시에 다른 부분을 터치할 경우 : 아무 반응이 없다
11	Location Inform	1. 기본: 결제할 상품이 판매되는 모든 자판기의 위치를 출력한다.	1. 기본: 다시 메시지를 요청한다.
12	Create Verification Code	1. 기본: 6자리 코드를 생성한다	1. 숫자가 아닌 다른 문자가 포함되는 경우 : 다시 생성한다.
13	Show Verification Code	1. 기본: 선구매 인증코드를 보여준다.	1. 기본: 아무 반응이 없다.
14	Item out	1. 기본: 결제한 음료를 제공한다.	1. 기본: 아무 반응이 없다.
15	Read Verification Code	1. 기본: 코드를 입력받는다. 2. 취소를 누를 경우: Select Mode로 돌아간다.	1. 기본: 아무 반응이 없다.
16	Check Verification Code	1. 기본:	1. 잘못된 코드인 경우:

		입력받은 코드를 검사하여 유효 여부를 반환한다.	다시 Code를 입력 받는다.
17	Reset Verification Code	1. 기본: 사용된 코드를 폐기하고, 모든 자판기에 코드 폐기 내용을 알린다.	1. 기본: 시스템이 재부팅 된다.
18	Message Request	1. 기본: 다른 자판기에게 정해진 규약에 따라 메시지를 보낸다.	1. 존재하지 않는 자판기의 네트워크 id에게 보낼 경우: Response Error를 받는다. 2. Response를 받지 못할 경우: Request를 다시 보낸다.
19	Message Response	1. 기본: 다른 자판기에게 정해진 규약에 따라 메시지를 보낸다.	1. 기본: 아무 반응이 없다.

- Non Functional Requirements
  - 직관적인 UI를 제공하고 사용하기 편해야 한다.
  - Java 사용
  - 네트워크 통신 간에 데이터 누락이 없어야 한다.
  - 단독망 사용
  - 네트워크 통신 간에 Latency 가 없어야 한다.
- Performance requirements
  - 자판기간 통신 응답 시간이 1초 이내로 수행되어야 한다.
  - 데이터를 확인하는 시간이 1초 이내로 수행되어야 한다.
- Operating Environment
  - Microsoft windows 10
- User Interface requirements
  - fresh store 자판기의 UI를 벤치마크한다.
  - 처음 사용하는 사람도 큰 문제없이 사용할 수 있어야 한다.
- Other requirements
  - 다른 자판기와 통신되지 않더라도 혼자 동작될 수 있어야 한다.
- Resources
  - Man Month : 5 Persons
    - A Team Leader
    - A Document Manager
    - 2-3 Engineers
  - Period : 6 Days(Around 20Hours)
  - Hardware : Intel PC

- Software
  - OS : Windows 10
  - Programming Language : JAVA (JDK 1.8)
  - Case Tool : draw.io , miro.com
- Configuration Management
  - Version Control System 을 활용
- Quality Assurance Plan
  - Iteration 마다 Technical Review를 할 계획이다.
  - Quality 체크 리스트를 통해 점검할 것이다.