

# Software Requirements Specification for Distributed Vending Machine

Team 2

201810286 박혜린

201811264 변장훈

201311312 정인욱

<b>1. Introduction</b>	<b>4</b>
1.1. Purpose	4
1.2. Scope	4
1.3. Definitions, Acronyms, and Abbreviations	4
1.4. References	5
1.5. Overview	5
<b>2. Overall description</b>	<b>5</b>
2.1. Product perspective	6
2.1.1. System interfaces	6
2.1.2. User interfaces	6
2.1.3. Hardware interfaces	6
2.1.5. Memory constraints	6
2.1.6. Operations	7
2.2. Product functions	7
2.3. User characteristics	7
2.4. Constraints	7
2.5. Assumptions and dependencies	8
<b>3. Specific requirements</b>	<b>9</b>
3.1. External interface requirements	9
3.1.1. User Interfaces	9
3.1.2. Hardware Interfaces	9
3.1.3. Software Interfaces	10
3.1.4. Communication Interfaces	10
3.2. System Features	10
3.2.1. Main Mode	10
3.2.1.1. Introduction/Purpose of feature	10
3.2.1.2. Stimulus/Response sequence	11
3.2.1.2.1 Sequence 1	11
3.2.1.2.2 Sequence 2	11
3.2.1.3. Associated functional requirements	11
3.2.1.3.1. Functional requirement 1	11
3.2.1.3.2. Functional requirement 2	11
3.2.2. Card Payment	11
3.2.2.1. Introduction/Purpose of feature	11
3.2.2.2. Stimulus/Response sequence	11
3.2.2.2.1 Sequence 1	11
3.2.2.3. Associated functional requirements	12
3.2.2.3.1. Functional requirement 1	12
3.2.3. Verification Code Mode	12
3.2.3.1. Introduction/Purpose of feature	12
3.2.3.2. Stimulus/Response sequence	12
3.2.3.2.1 Sequence 1	12

3.2.3.2.2 Sequence 2	12
3.2.3.3. Associated functional requirements	13
3.2.3.3.1. Functional requirement 1	13
3.2.3.3.2. Functional requirement 2	13
3.2.4. Prepay	13
3.2.4.1. Introduction/Purpose of feature	13
3.2.4.2. Stimulus/Response sequence	13
3.2.4.2.1 Sequence 1	13
3.2.4.2.2 Sequence 2	13
3.2.4.2.3 Sequence 3	13
3.2.4.2.4 Sequence 4	13
3.2.4.2.5 Sequence 5	14
3.2.4.2.6 Sequence 6	14
3.2.4.3. Associated functional requirements	14
3.2.4.3.1. Functional requirement 1	14
3.2.4.3.2. Functional requirement 2	14
3.2.4.3.3. Functional requirement 3	14
3.2.4.3.4. Functional requirement 4	15
3.2.4.3.5. Functional requirement 5	15
3.2.4.3.6. Functional requirement 6	15
3.2.5. Refund	15
3.2.5.1. Introduction/Purpose of feature	15
3.2.5.2. Stimulus/Response sequence	15
3.2.5.2.1. Sequence 1	15
3.2.5.3. Associated functional requirements	15
3.2.5.3.1. Functional requirement 1	15
3.2.6. Admin Mode	15
3.2.6.1. Introduction/Purpose of feature	16
3.2.6.2. Stimulus/Response sequence	16
3.2.6.2.1. Sequence 1	16
3.2.6.2.2. Sequence 2	16
3.2.6.3. Associated functional requirements	16
3.2.6.3.1. Functional requirement 1	16
3.2.6.3.2. Functional requirement 2	16
3.2.7. Start Mode	16
3.2.7.1. Introduction/Purpose of feature	16
3.2.7.2. Stimulus/Response sequence	17
3.2.7.2.1 Sequence 1	17
3.2.7.2.2 Sequence 2	17
3.2.7.2.3 Sequence 3	17
3.2.7.3. Associated functional requirements	17
3.2.7.3.1. Functional requirement 1	17
3.2.7.3.2. Functional requirement 2	17
3.3. Performance requirements	18

3.4. Design constraints	18
3.5. Software system attributes	18

# 1. Introduction

## 1.1. Purpose

본 문서는 분산자판기(DVM : Distributed Vending Machine) 시스템의 자판기 controller software를 구현하기 위한 요구사항을 명시한다.

## 1.2. Scope

DVM은 사용자가 구매한 음료를 제공하는 자동 판매기로 다른 자판기들과 네트워크로 연결되어 있어 현재 이 자판기에서 판매하지 않지만 다른 자판기에서 판매중인 음료에 대한 정보를 줄 수 있으며 사용자가 원한다면 선결제 방식을 통해 다른 자판기의 음료를 미리 구매한 후 인증코드를 받는다. 그 후 사용자가 구매한 음료를 판매하는 자판기에서 인증코드를 입력하면 음료를 받을 수 있다. 본 문서에서 서술하는 DVM controller software는 DVM이 기능대로 동작할 수 있도록 하는 소프트웨어이다.

## 1.3. Definitions, Acronyms, and Abbreviations

Num	Term	Description
1	SW	Software의 약자
2	DVM	Distributed Vending Machine의 약자로 네트워크로 연결된 분산 자판기 중 하나를 지칭한다.
3	선결제	사용자가 이용하고 있는 DVM에서 판매 불가능한 메뉴를 다른 DVM에서 구매하고자 할 때 이용하는 결제 시스템이다. 사용자는 음료를 미리 구매하고 인증코드를 받아 해당 자판기에서 권리를 행사할 수 있다.
4	Verification Code	사용자가 선결제를 한 경우 음료 구매권을 행사하기 위한 인증 코드이다. '0'부터 '9'까지 10자리의 숫자로 구성된 길이가 4인 String이다.

5	Stock	음료들의 재고
6	Address	DVM이 설치된 주소로 길이가 1 이상 10 이하인 String이다.
7	Msg	Message의 줄임말로 DVM이 네트워크 상으로 연결된 다른 DVM에게 전송하거나 전송 받는 데이터 단위. Message Protocol에 의해 형식이 정의된다.
8	Broadcast	네트워크로 연결된 모든 DVM(작동 중인 DVM과 작동 중이지 않은 DVM을 모두 포함한다)에게 메시지를 전송하는 방식
9	Admin Code	인증코드 입력화면에서 관리자 화면으로 진입하기 위한 코드이다. '0'부터 '9'까지 10자리의 숫자로 구성된 길이가 6인 String이다.
10	VMList	네트워크상에 작동 중인 자판기들의 목록
11	ID	자판기 고유의 번호
12	CardList	카드 번호, 유효성, 잔액을 저장한 배열

## 1.4. References

- IEEE Standard 830-1998
- [Microsoft Word - Distributed Vending Machine.docx \(konkuk.ac.kr\)](#)
- [dslab.konkuk.ac.kr/Class/2021/21SMA/Lecture Note/OOPT\\_All\(2020\).pdf](#)

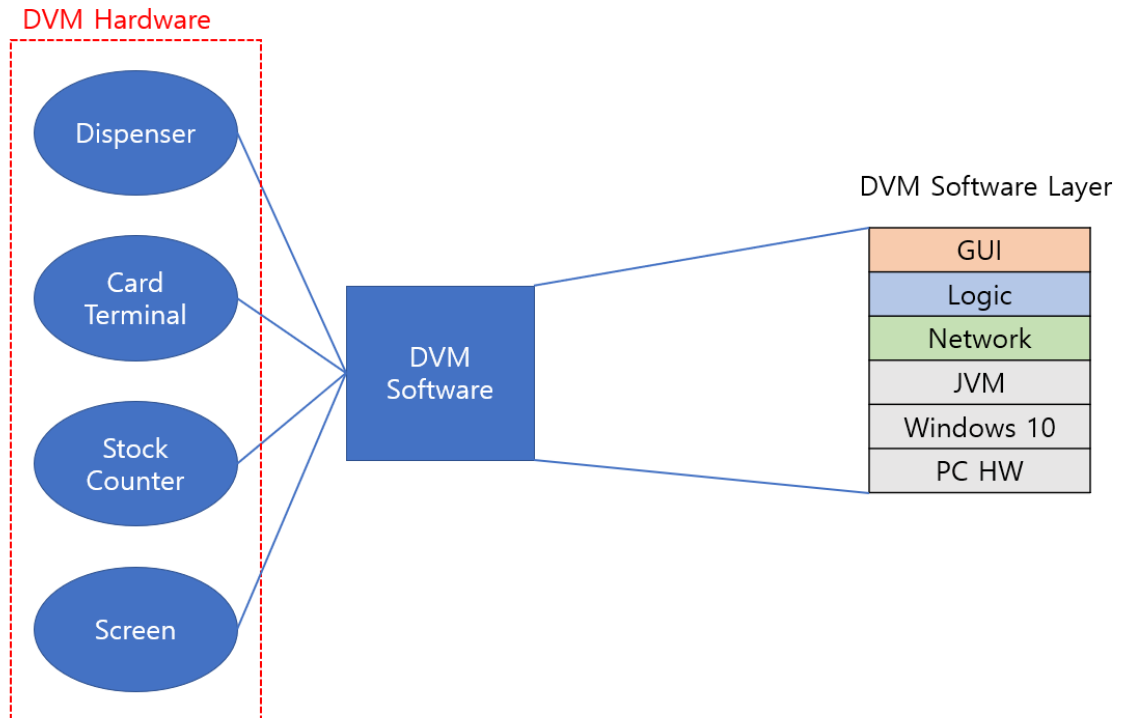
## 1.5. Overview

Chapter 2, Overall Description에서는 이 SW의 전반적인 기능에 대해 설명하며, 다음 Chapter의 requirements specification의 기본 요소를 정리한다. Chapter 3, Specific Requirements에서는 이 SW의 개발을 위한 설명으로 기능들의 자세한 사항들과 용어를 정리한다.

## 2. Overall description

## 2.1. Product perspective

### 2.1.1. System interfaces



### 2.1.2. User interfaces

관리자는 자판기를 시작하기 전에 자판기 ID, 자판기 주소, 관리자 모드 코드  
 사용자는 디스플레이상에서 버튼 입력 방식으로 음료, 자판기를 선택할 수 있다.  
 사용자는 디스플레이상에서 버튼 입력 방식으로 인증코드, 시작, 취소를 선택하여  
 해당 기능을 실행할 수 있다.

### 2.1.3. Hardware interfaces

DVM SW와 연결된 HW로는 Screen, Stock Counter, Dispenser, Credit Card Terminal가 있다. 서술한 HW들은 DVM 내부에 각각의 개별적인 interface와 연결되어 있으며 이는 3.1.2절에서 설명한다.

### 2.1.5. Memory constraints

DVM SW가 필요로 하는 메모리 공간은 Screen 출력 화면 등의 내부 변수 저장 공간을 포함하여 모두 32MB를 넘지 않는 것으로 한다.  
 각 변수의 메모리 사용량은 모두 정해져있다.

### 2.1.6. Operations

DVM은 네트워크에 미리 연결되어 있고 관리자가 자판기 ID, Address, Admin Code를 파일에 저장하고 DVM을 시작시키면 사용자가 음료를 구매할 수 있는 판매상태로 작동한다. 전원이 끊겨 비정상적으로 종료되는 경우 이외에는 관리자가 DVM을 종료할 때까지 작동한다.

## 2.2. Product functions

- 하나의 DVM은 7가지 음료만 재고로 가질 수 있으며, 재고로 가지고 있지 않지만 선결제가 가능한 13가지의 음료를 포함해 총 20가지의 음료를 판매한다.
- 사용자가 DVM내에 재고가 있는 음료를 카드로 구매할 경우 해당 음료를 제공한다.
- 선결제 기능은 다음과 같다.
  - 사용자가 DVM내에 재고가 없는 음료를 선택할 경우 네트워크 통신으로 VMList 중 해당 음료의 재고가 있는 DVM들을 사용자에게 보여준다.
  - 사용자가 재고가 있는 DVM들 중 하나를 선택하면 선결제를 위해 카드 입력을 요구한다.
  - 사용자가 해당 음료를 카드로 선결제 할 경우 현재 DVM에서 인증코드를 발급한다.
  - VMList에서 하나의 자판기를 선택하는 사이 사용자가 선택한 DVM의 음료 재고가 0이 될 경우 결제한 금액을 환불한다.
  - 사용자는 자신이 선택한 DVM에서 인증코드를 입력하여 음료를 제공할 수 있다.

## 2.3. User characteristics

원하는 음료를 구매하고 싶은 사용자를 타겟으로 한다. 사용자는 디스플레이를 활용하여 자신이 원하는 음료를 선택할 수 있고 카드 단말기가 읽을 수 있는 신용카드를 보유해야 한다. 또한 단말기에 읽히는 신용카드는 구매한 음료에 대한 금액 이상의 잔액이 있어야 한다.

## 2.4. Constraints

- 다른 DVM으로부터 전송받는 Msg는 수신 즉시 처리한다.
- 동시에 Msg를 받는 경우는 없다고 가정한다.
- 전송된 네트워크 Msg가 작동 중인 DVM에 도달하지 않는 경우는 없다고 가정한다.
- 터치 패드에서 한번에 두개 이상의 입력이 동시에 들어올 수 없다.
- DVM 내 하나의 음료의 재고 최대값은 100이다.
- 하나의 DVM은 Verification Code는 최대 1,000개까지 저장할 수 있다.
- 카드의 잔액은 2,147,483,647이하여야 한다.
- 각 음료의 가격은 0 초과 20,000 이하여야 한다.
- 자판기 주소의 길이는 0 초과 10 이하인 String이어야 한다.

- 카드의 번호는 ‘-’를 제외한 ‘0’부터 ‘9’까지의 10가지 숫자로 구성된 12자리 String이다.

## 2.5. Assumptions and dependencies

- SW 내부에 카드 정보와 유효성, 잔액이 CardList에 저장된다. 현재 SW의 카드 결제 시스템은 이 정보를 이용하여 결제가 이루어진다.
- 사용자가 입력한 카드의 카드 번호가 CardList에 없다면 유효하지 않은 카드로 가정한다.
- 하나의 DVM은 7가지 음료를 판매한다.
- 메시지는 수신 즉시 처리되는 것으로 가정하며, 동시에 두 대 이상의 DVM에서 메시지를 수신하는 경우는 없다고 가정한다.
- DVM은 최대 10개 까지 연결 가능하다.
- 각 DVM은 1부터 10까지의 중복 되지 않는 고유의 ID를 가지고 있다.
- 각 자판기는 모두 네트워크에 연결되어 있고 네트워크 연결 정보는 미리 알고 있다고 가정한다.
- 메시지 처리 속도가 충분히 빨라 VMList의 모든 자판기에게 재고 여부 요청을 하고 응답을 받기 전 VMList 외의 자판기가 시작하는 경우는 없다고 가정한다.
- 자판기의 판매 음료 종류는 사전에 결정된다.
- 음료의 ID와 이름은 다음의 20가지이다.

음료 ID	음료 이름
0	코카콜라
1	제로콜라
2	칠성사이다
3	초록매실
4	제주감귤
5	파워에이드
6	게토레이
7	레쓰비
8	포카리스웨트
9	햇식스
10	환타
11	몬스터
12	코코팜
13	갈아만든 배



14	레모네이드
15	웰치스
16	토레타
17	쿠우
18	썩썩
19	T.O.P

## 3. Specific requirements

### 3.1. External interface requirements

#### 3.1.1. User Interfaces

- 사용자는 취소 버튼이 있는 화면에서 취소 버튼을 선택하면 메뉴 선택 대기 화면으로 이동한다.
- 사용자는 0부터 9까지의 숫자 버튼들과 지우기 버튼을 이용하여 인증코드 입력 기능에서 인증코드를 입력할 수 있다.
- 사용자는 메뉴 선택 대기 화면에서 20 종류의 메뉴 버튼 중 하나를 선택하여 음료를 선택할 수 있다.  
재고가 있는 메뉴를 선택 시 카드결제가 진행되며, 재고가 없는 메뉴를 선택 시 재고가 있는 다른 DVM을 선택해 선결제를 진행할 수 있다.
- 사용자는 재고가 있는 VMList에서 해당 자판기 버튼을 선택하여 다른 자판기를 선택할 수 있다.
- 메뉴 선택 대기 화면에서 인증코드 버튼을 선택하여 인증코드 입력 진입 기능을 실행할 수 있다.
- 관리자는 시작 버튼을 선택하여 다른 DVM들에게 시작 메시지를 보낼 수 있다.

#### 3.1.2. Hardware Interfaces

번호입력과 확인, 메시지를 보여주는 스크린 하드웨어 인터페이스, 카드 결제를 위한 하드웨어 인터페이스가 존재한다.

Hardware interface	Description
--------------------	-------------

Screen interface	Screen에서 들어온 사용자의 입력을 처리하여 SW로 전달하거나 사용자에게 필요한 정보를 Screen으로 전달하여 출력할 수 있게한다.
Credit card terminal interface	사용자의 카드정보를 입력받아 Credit Card Payment Function을 통해 결제가 이루어질 수 있도록 한다.
Stock Counter interface	SW에서 재고 확인 요청이 발생하면 Stock Counter를 작동시켜 재고를 확인하고 정보를 SW로 넘겨준다.
Dispenser interface	구매가 이루어졌을때 사용자가 구매한 음료에 대한 정보를 SW로 부터 받아 Dispenser가 출력할 수 있도록 한다.

### 3.1.3. Software Interfaces

JDK 15버전에서 실행한다.

### 3.1.4. Communication Interfaces

- Message Protocol은 다음과 같이 정의된다.

src_id	dst_id	msg type	msg description(optional)
		Msg 타입	Msg 내용 (타입에 따라 결정)
현재 자판기 ID	대상 자판기 ID (broadcast는 0)	1. 자판기 시작 요청 및 주소	Address
		2. 자판기 시작 응답 및 주소	Address
		3. 자판기 종료	Null
		4. 재고 여부 요청	음료 번호
		5. 재고 여부 응답	해당 음료의 재고 유/무
		6. 인증코드 생성 요청	음료 번호
		7. 인증코드 응답	인증코드
		8. 환불 요청	Null

## 3.2. System Features

### 3.2.1. Main Mode

#### 3.2.1.1. Introduction/Purpose of feature

자판기의 메뉴 선택 대기 화면이다.

사용자로부터 메뉴를 입력 받거나, 인증코드 입력 모드로 전환한다.

### 3.2.1.2. Stimulus/Response sequence

#### 3.2.1.2.1 Sequence 1

- **Stimulus**  
사용자가 메뉴를 선택한다.
- **Response**  
사용자가 선택한 메뉴의 재고가 있을 경우 3.2.2의 Card Payment로 이동한다.  
사용자가 선택한 메뉴의 재고가 없을 경우 3.2.4의 PrePay로 이동한다.

#### 3.2.1.2.2 Sequence 2

- **Stimulus**  
사용자가 인증코드 입력 모드로 전환 버튼을 누른다.
- **Response**  
3.2.2의 Verification Mode로 이동한다.

### 3.2.1.3. Associated functional requirements

#### 3.2.1.3.1. Functional requirement 1

- 사용자에게 20가지 메뉴의 가격과 이름이 적힌 버튼을 표시한다.

#### 3.2.1.3.2. Functional requirement 2

- 사용자에게 인증코드 모드로 전환할 버튼을 표시한다.

## 3.2.2. Card Payment

### 3.2.2.1. Introduction/Purpose of feature

사용자가 선택한 음료의 재고가 있을 경우 사용자에게 카드 결제를 요구한다.

### 3.2.2.2. Stimulus/Response sequence

#### 3.2.2.2.1 Sequence 1

- **Stimulus**  
사용자가 결제할 카드를 카드 단말기에 입력한다.
- **Response**  
입력 받은 카드의 유효성을 검사한다.  
유효하지 않은 카드일 경우 5초간 에러 메시지를 출력하고 3.2.1의 Main Mode로 이동한다.

입력 받은 카드의 잔액이 선택한 메뉴의 금액 이상인지 검사한다.  
잔액이 부족한 경우 5초간 에러 메시지를 출력하고 3.2.1의 **Main Mode**로 이동한다.  
사용자가 선택한 메뉴의 금액만큼 입력 받은 카드의 잔액을 차감한다.  
사용자가 선택한 음료를 제공한 후 3.2.1의 **Main Mode**로 이동한다.

### 3.2.2.3. Associated functional requirements

#### 3.2.2.3.1. Functional requirement 1

- **DVM**은 카드의 번호, 유효성, 잔액이 저장된 리스트(**CardList**)를 가지고 있다.
- 사용자가 입력한 카드가 **CardList**에 없거나 혹은 **CardList**에 있으나 유효하지 않다면 결제되지 않아야한다.
- 사용자가 입력한 카드의 잔액이 사용자가 선택한 음료보다 작을 경우 결제되지 않아야한다.
- 카드가 정상 결제된 경우 사용자가 선택한 음료가 제공되어야 한다.

### 3.2.3. Verification Code Mode

#### 3.2.3.1. Introduction/Purpose of feature

인증코드 입력 대기 모드이다.  
사용자로부터 인증코드를 입력 받아 음료를 제공하거나, 관리자로부터 관리자 코드를 입력 받아 관리자 모드로 전환한다.

#### 3.2.3.2. Stimulus/Response sequence

##### 3.2.3.2.1 Sequence 1

- **Stimulus**  
사용자가 인증코드(4자리 숫자)를 입력한다.
- **Response**  
인증코드의 유효성을 검사한다.  
유효한 인증코드일 경우 제품을 제공한다.  
유효하지 않은 인증코드일 경우 에러 메시지를 5초간 출력한 후 3.2.1의 **Main Mode**로 이동한다.

##### 3.2.3.2.2 Sequence 2

- **Stimulus**  
관리자가 관리자 코드(6자리 숫자)를 입력한다.
- **Response**  
**Admin Code**와 입력한 코드가 일치할 경우 3.2.6의 **Admin Mode**로 이동한다.  
일치하지 않을 경우 에러 메시지를 5초간 출력한 후 3.2.1의 **Main Mode**로 이동한다.

### 3.2.3.3. Associated functional requirements

#### 3.2.3.3.1. Functional requirement 1

- 사용자가 입력한 인증코드가 유효할 경우 사용자가 선결제 했던 음료를 제공해야 한다.
- 4자리의 인증코드에 대해서만 유효성을 검사한다.

#### 3.2.3.3.2. Functional requirement 2

- 관리자가 입력한 인증코드가 **Admin Code**와 일치하면 관리자 모드로 이동해야 한다.

## 3.2.4. Prepay

### 3.2.4.1. Introduction/Purpose of feature

사용자가 구매하고자 하는 음료의 재고가 없을 경우 사용자에게 판매 가능한 자판기의 목록을 보여주며 선결제 여부를 요구한다. 사용자가 목록의 자판기 중 특정 자판기를 선택하면 해당 자판기에 대해서 선결제를 진행하는 것으로 본다.

### 3.2.4.2. Stimulus/Response sequence

#### 3.2.4.2.1 Sequence 1

- **Stimulus**  
다른 자판기로부터 재고 여부 메시지(Msg type 4)를 응답한다.
- **Response**  
해당 메뉴의 재고를 확인한 후 재고 여부 응답(Msg type 5)을 발송한다.

#### 3.2.4.2.2 Sequence 2

- **Stimulus**  
다른 자판기로부터 재고 여부 응답(Msg type 5)을 수신한다.
- **Response**  
VMList의 모든 자판기로부터 응답을 받은 뒤, 재고가 있다고 응답한 자판기의 목록을 사용자에게 보여준다.

#### 3.2.4.2.3 Sequence 3

- **Stimulus**  
사용자가 자판기 목록에서 자판기를 선택한다.
- **Response**  
사용자가 자판기 목록에서 자판기를 선택한 경우 선결제 요구를 한다.  
3.2.4.2.4. Sequence 4로 이동한다.

#### 3.2.4.2.4 Sequence 4

- **Stimulus**

사용자가 선결제할 카드를 입력한다.

- **Response**

입력 받은 카드의 유효성을 검사한다.

유효하지 않은 카드일 경우 5초간 에러 메시지를 출력하고 3.2.1의 Main Mode로 이동한다.

입력 받은 카드의 잔액이 선택한 메뉴의 금액 이상인지 검사한다.

잔액이 부족한 경우 5초간 에러 메시지를 출력하고 3.2.1의 Main Mode로 이동한다.

사용자가 선택한 메뉴의 금액만큼 입력 받은 카드의 잔액을 차감한다.

사용자가 선택한 자판기에 Verification Code 생성 요청(Msg type 6)을 발송한다.

#### 3.2.4.2.5 Sequence 5

- **Stimulus**

다른 자판기로부터 Verification Code 요청(Msg type 6)을 수신한다.

- **Response**

해당 제품의 재고가 있을 경우 현재 DVM 내부에 저장된 Verification Code와 중복되지 않는 Verification Code를 생성한다.

생성된 Verification Code(Msg type 7)를 발송한다.

해당 제품의 재고가 없을 경우 환불 요청(Msg type 8)을 발송한다.

#### 3.2.4.2.6 Sequence 6

- **Stimulus**

다른 자판기로부터 생성된 Verification Code(Msg type 7)을 수신한다.

- **Response**

수신한 Verification Code를 30초 동안 화면에 출력한다. 30초가 지나거나 30초가 지나기 전에 취소 버튼이 선택될 경우 3.2.1의 Main Mode로 이동한다.

### 3.2.4.3. Associated functional requirements

#### 3.2.4.3.1. Functional requirement 1

- 자판기가 모든 상태에서 다른 자판기의 재고 여부 메시지를 수신할 수 있어야 한다.
- 자판기가 음료에 따른 자신의 재고가 저장된 컬렉션(Stock)을 가지고 있다.

#### 3.2.4.3.2. Functional requirement 2

- 자판기 목록은 (0~10)개 까지 출력 가능하다.

#### 3.2.4.3.3. Functional requirement 3

- Screen에 음료의 재고가 있다고 응답한 자판기의 목록이 모두 표시되어야 한다.
- 사용자가 자판기를 선택하면 카드 결제를 진행한다.

#### 3.2.4.3.4. Functional requirement 4

- DVM은 **CardList**를 가지고 있다.
- 사용자가 입력한 카드가 **CardList**에 없거나 혹은 **CardList**에 있으나 유효하지 않다면 결제되지 않아야한다.
- 사용자가 입력한 카드의 잔액이 사용자가 선택한 음료의 금액보다 작을 경우 결제되지 않아야한다.

#### 3.2.4.3.5. Functional requirement 5

- **Verification Code**는 랜덤한 숫자로 생성되며, 새로운 **Verification Code**를 생성할 경우 이미 생성되어 저장하고 있으며 아직 사용되지 않은 **Verification Code**와 중복되지 않아야 한다.

#### 3.2.4.3.6. Functional requirement 6

- 다른 자판기로부터 수신한 **Verification Code**를 사용자가 충분히 식별할 수 있게 출력한다.
- 사용자가 **Verification Code**를 충분히 기억할 수 있는 시간으로 30초를 주어야 한다.

### 3.2.5. Refund

#### 3.2.5.1. Introduction/Purpose of feature

자판기가 다른 자판기로부터 환불 요청을 받아 환불을 실행하는 기능이다.

#### 3.2.5.2. Stimulus/Response sequence

##### 3.2.5.2.1. Sequence 1

- **Stimulus**  
자판기가 다른 자판기로부터 환불 요청 메시지(**Msg Type 8**)를 수신한다.
- **Response**  
자판기가 환불을 실행하고 에러 메시지를 5초간 출력한 후 **3.2.1. Main Mode**로 이동한다.

#### 3.2.5.3. Associated functional requirements

##### 3.2.5.3.1. Functional requirement 1

- 자판기가 **Request Verification Code**를 보낸 후 메시지 수신 대기 상태에서 다른 자판기의 환불 요청 메시지를 수신할 수 있어야 한다.
- **Verification Code** 생성에 실패한 해당 선결제에 대해 **CardList**의 카드잔액을 결제 전의 값으로 수정해야 한다.

## 3.2.6. Admin Mode

### 3.2.6.1. Introduction/Purpose of feature

관리자가 자판기의 음료 재고 관리 및 종료를 수행하는 기능이다.

### 3.2.6.2. Stimulus/Response sequence

#### 3.2.6.2.1. Sequence 1

- **Stimulus**  
관리자가 재고관리 기능을 실행한다.
- **Response**  
자판기의 재고 측정 센서가 자신의 재고를 측정해 자신의 재고가 저장된 컬렉션(Stock)에 수량을 반영한다.

#### 3.2.6.2.2. Sequence 2

- **Stimulus**  
관리자가 자판기 종료를 선택한다.
- **Response**  
자판기가 다른 자판기에 종료 메시지(Msg Type 3)를 전송한다.
- **Stimulus**  
자판기가 다른 자판기의 종료 메시지(Msg Type 3)에 응답한다.
- **Response**  
자판기가 VMList에서 종료 메시지(Msg Type 3)를 보낸 해당 자판기를 제거한다.

### 3.2.6.3. Associated functional requirements

#### 3.2.6.3.1. Functional requirement 1

- 재고 측정 센서가 정상작동해야 한다.
- 기존의 재고를 새로 측정된 재고로 최신화한다.

#### 3.2.6.3.2. Functional requirement 2

- 자판기가 모든 상태에서 다른 자판기의 종료 메시지 수신가능 해야 한다
- 자판기가 다른 모든 자판기에 종료 메시지를 전송해야 한다.

## 3.2.7. Start Mode

### 3.2.7.1. Introduction/Purpose of feature

관리자가 자판기의 정보를 입력하고 자판기의 작동을 시작한다.



### 3.2.7.2. Stimulus/Response sequence

#### 3.2.7.2.1 Sequence 1

- **Stimulus**  
관리자가 자판기의 ID, Address, Admin Code를 파일에 입력한 후 자판기를 시작한다.
- **Response**  
Broadcast로 자판기 시작 메시지(Msg Type 1)를 요청한다.

#### 3.2.7.2.2 Sequence 2

- **Stimulus**  
다른 자판기로부터 자판기 시작 메시지(Msg Type 1)를 수신
- **Response**  
해당 자판기를 VMList에 추가 후 그 자판기에 자판기가 작동 중임을 알리는 메시지(Msg Type 2)를 전송한다.

#### 3.2.7.2.3 Sequence 3

- **Stimulus**  
자판기 작동 메시지(Msg Type 2)를 수신
- **Response**  
수신된 자판기들의 ID와 주소를 VMList에 저장한다.  
자판기 재고를 측정 후 메뉴 선택 대기화면을 출력

### 3.2.7.3. Associated functional requirements

#### 3.2.7.3.1. Functional requirement 1

- 유효한 자판기 정보는 다음과 같다.
  - ID : 1 이상 10 이하의 자연수
  - Address : 길이가 1 이상 10 이하인 String
  - Admin Code : 길이가 6인 '0'부터 '9'까지의 숫자로만 구성된 String
- 유효하지 않은 자판기 정보를 파일에 입력한 뒤 실행시키면, 즉시 종료한다.
- 자판기가 Broadcast로 시작 메시지(Msg Type 1)를 전송해야 한다.

#### 3.2.7.3.2. Functional requirement 2

- 자판기가 작동 중임을 알리는 메시지(Msg Type 2) 발송하는 시점에 VMList를 가지고 있어야 한다.

#### 3.2.7.3.3. Functional requirement 3

- 재고 측정 센서가 정확한 음료의 수량을 측정한 후 재고를 업데이트 한다.
- 자판기가 VMList를 가지고 있어야 한다.

### 3.3. Performance requirements

- 네트워크 설정 시간을 포함하여 자판기를 시작하고 메인 메뉴가 실행하는데 걸리는 시간은 **30s** 이하여야 한다.
- 작동 중인 네트워크 메시지 전송 속도는 **1s** 이하여야 한다.
- 사용자가 스크린에 입력 후 해당 기능 실행 시작까지 걸리는 **response time**은 모두 **0.5s** 이하여야 한다.
- 스크린 상에서 중복 선택이 불가능하게 한다.

### 3.4. Design constraints

- Java로 구현되어야 한다.
- Analysis, Design 방법은 OOPT 개발 방법을 준수한다.
- 제공받은 Continuous Testing and Integration Platform에서 제공받은 IDE를 사용하여야 한다.

### 3.5. Software system attributes

- **Reliability**  
자판기의 본래 목적에 충실하게 네트워크 상의 지연이 없어야한다.
- **Security**  
카드 정보를 외부에서 접근 가능하지 않게 한다.  
네트워크로 전송되는 인증코드 메시지가 연결된 DVM이외에서 접근 가능하지 않게 한다.
- **Usability**  
사용자와 시스템간 소통의 방식을 최소화하여 사용자의 잘못된 조작이 시스템 동작을 방해하지 않도록 한다.