

SDS

201511238 허준호
201911147 강속영
201711320 오준엽
201511218 이지영
201710397 이지훈

Index

Change Log

1.Introduction

1.1. Purpose

1.2. scope

1.3. Definitions,Acronyms, and Abbreviations

1.4. References

1.5. Overview

2. System architecture description

3. Object Oriented Design

3.1 Sequence Diagram

3.1.1. Start Order

3.1.2. Input Choice

3.1.3. CheckChosenItemStock & 3.1.4 Answer Chosen Stock

Info

3.1.5. Determine Location

3.1.6. Determine Prepayment

3.1.7. Payment

3.1.8. Create Precode

3.1.9. Complete Prepayment

3.1.10. Serve Item

3.1.11. Check Precode

3.1.12. Answer Precode Info

3.1.13. Input Precode

3.1.14. Enter Administrator Mode

3.1.15. Add Itemlist

3.1.16. Remove Itemlist

3.1.17. Change Stock

3.1.18. Change Item Price

3.1.19. Check Sales History

4. Class Diagram

5. Traceability Analysis

Change Log

class diagram

Precode

system architecture

paymentlineitem, card 삭제

1.Introduction

1.1. Purpose

본 문서의 목적은 분산자판기(DVM : Distributed Vending Machine) 시스템의 자판기 controller software를 개발하기 위함이다.

1.2. scope

사용자가 요구하는 음료가 현재 DVM에 존재하지 않을 경우 다른 DVM과의 통신을 통해 해당 음료의 재고를 확인하고, 사용자가 다른 기기에서 해당 음료를 제공받을 수 있게 해주는 인증코드를 발급한 후 인증코드를 재고가 있는 DVM에 입력시 그 음료를 수령할 수 있게 하는 소프트웨어이다.

1.3. Definitions,Acronyms, and Abbreviations

DVM: Distributed vending machine의 약자이며 한글로 분산 자판기 라고 한다. 이는 추후 설계 및 조립으로 구현된 구체적인 제품을 가리키는 말이기도 하다.

상품: DVM에서 판매하는 20가지 음료

prepayment: 선결제

precode: 선결제시 사용자에게 제공되는 코드

stock : DVM에 존재하는 상품(음료)의 재고

1.4. References

IEEE Std 830의 서술 방식을 참조하여 본 문서를 작성하였다.

Preliminary functional requirements Distributed vending machine(유준범 2021.03)

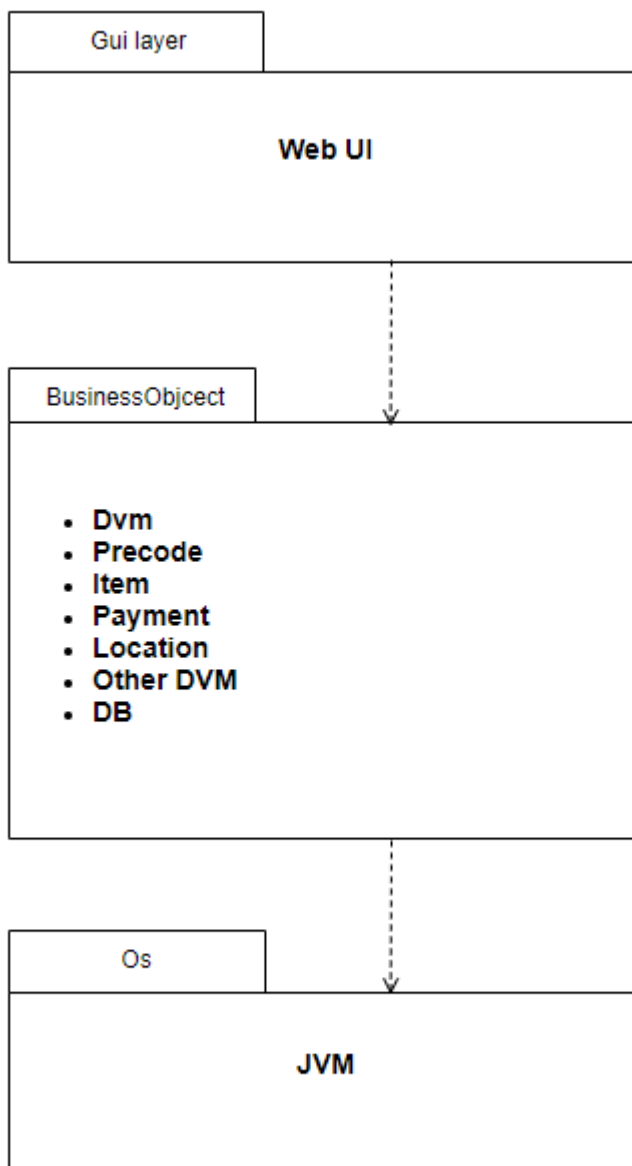
OOPT_All(2020)(유준범 2021.03)

1.5. Overview

2장에선 DVM 제품들의 관계, 그리고 DVM 소프트웨어와 하드웨어의 기능을 간략하게 제시할 것이다.

3장은 2장에서 제시한 DVM의 기능 및 인터페이스를 상세하게 설명하는 문단이다.

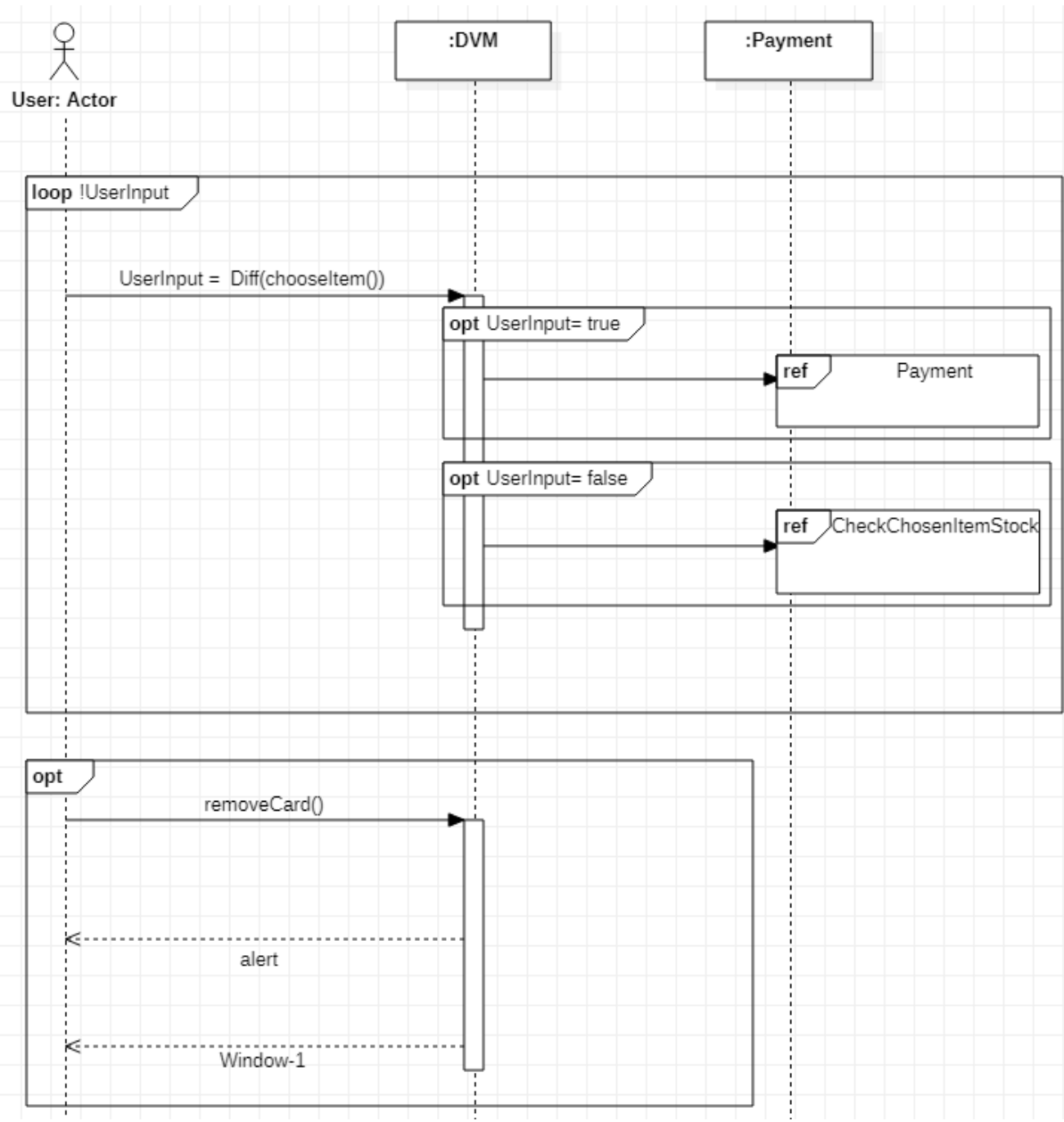
2. System architecture description



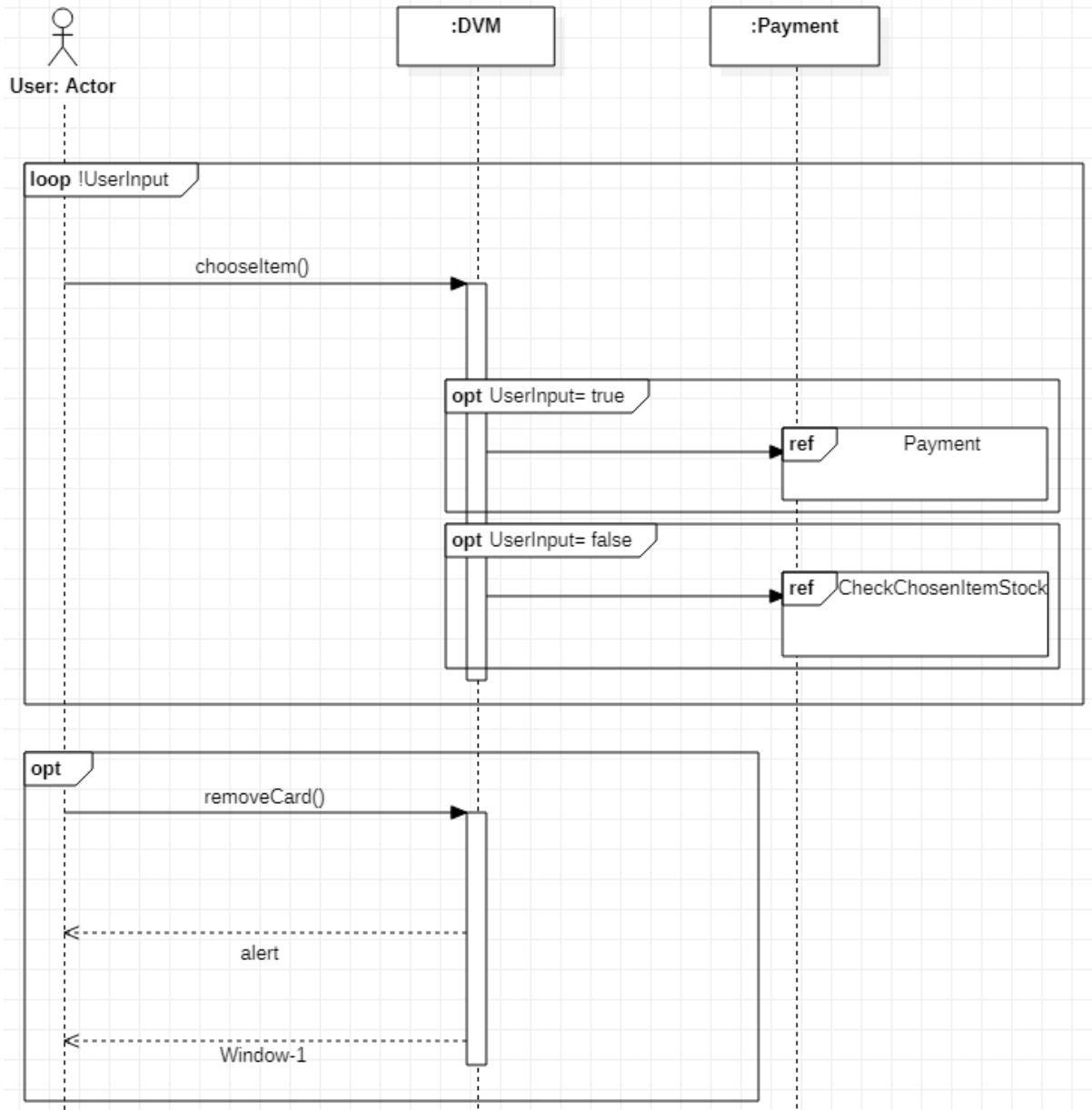
3. Object Oriented Design

3.1 Sequence Diagram

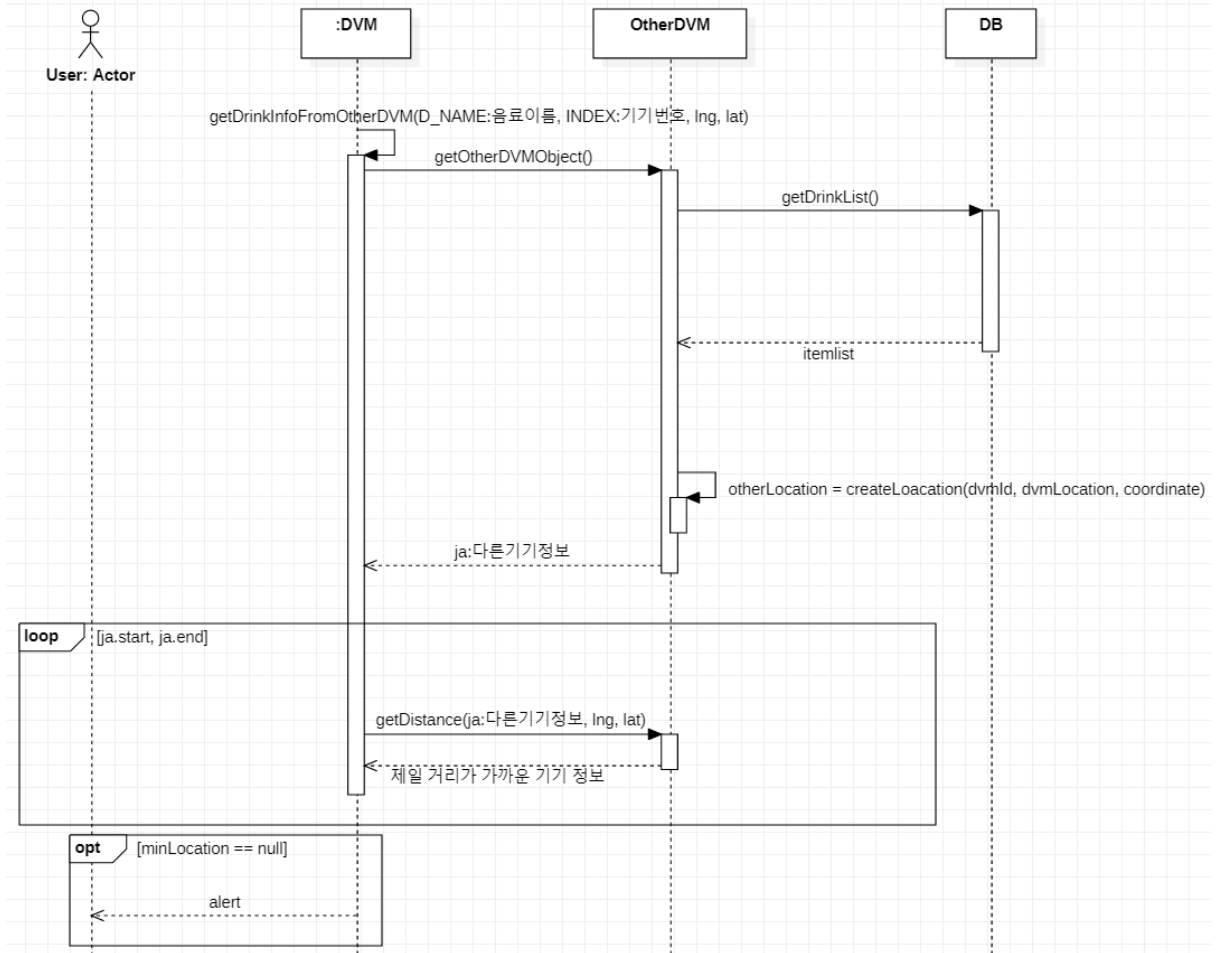
3.1.1. Start Order



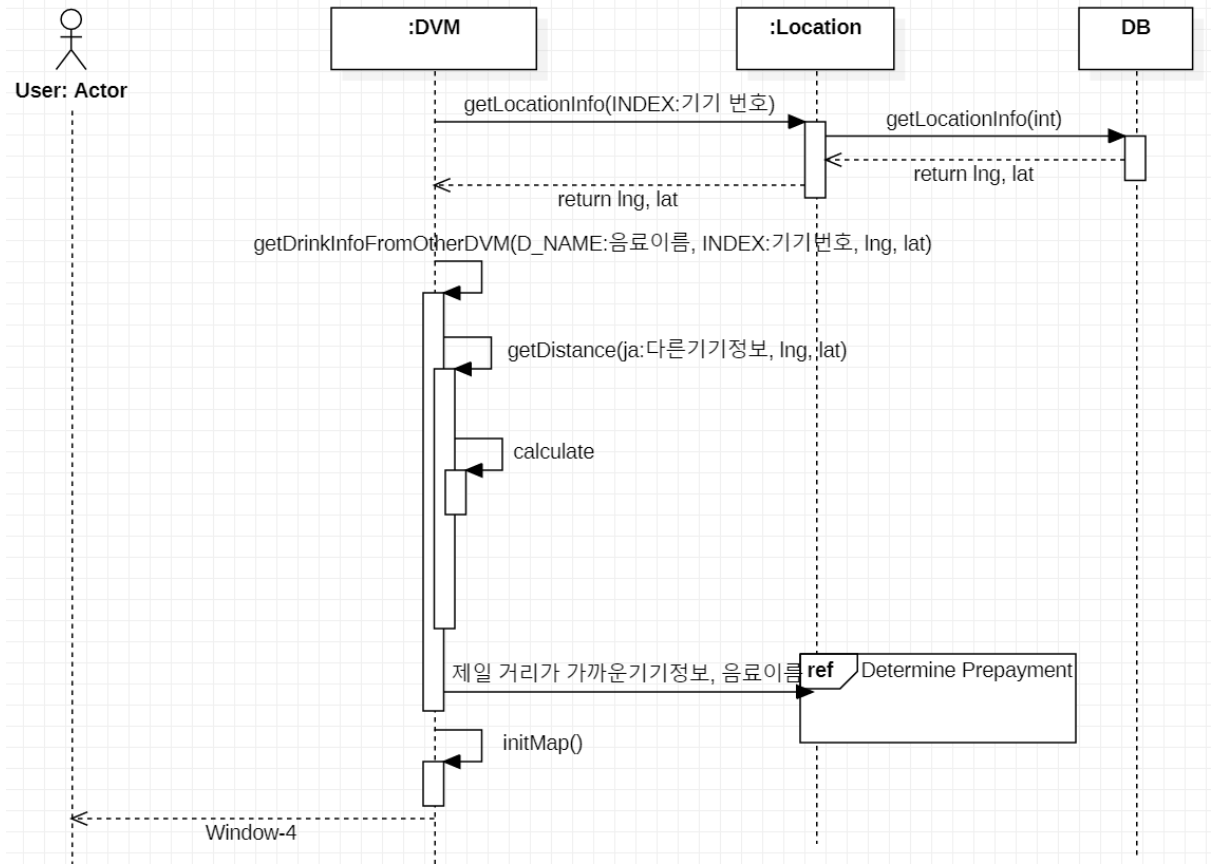
3.1.2. Input Choice



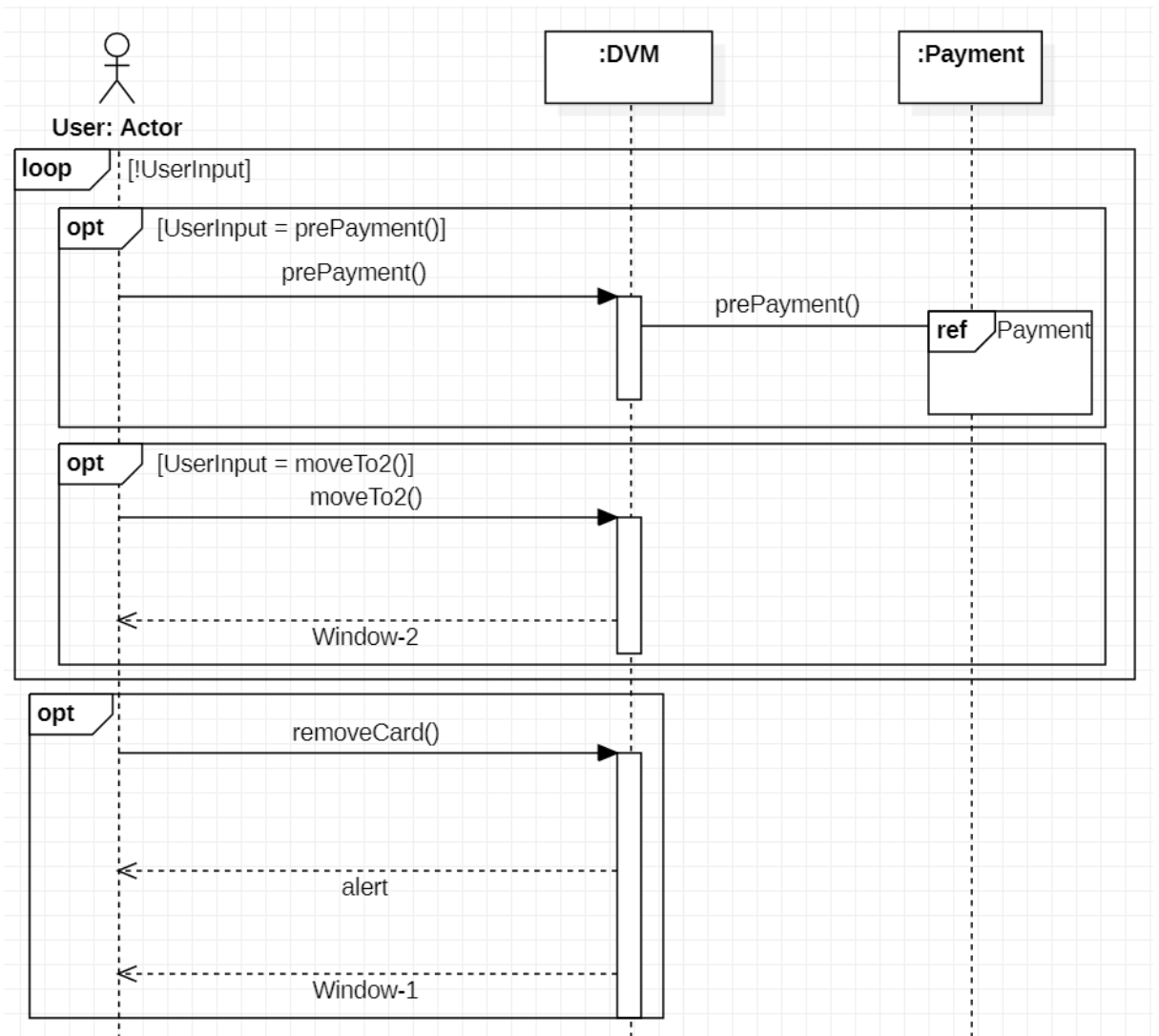
3.1.3. CheckChosenItemStock & 3.1.4 Answer Chosen Stock Info



3.1.5. Determine Location

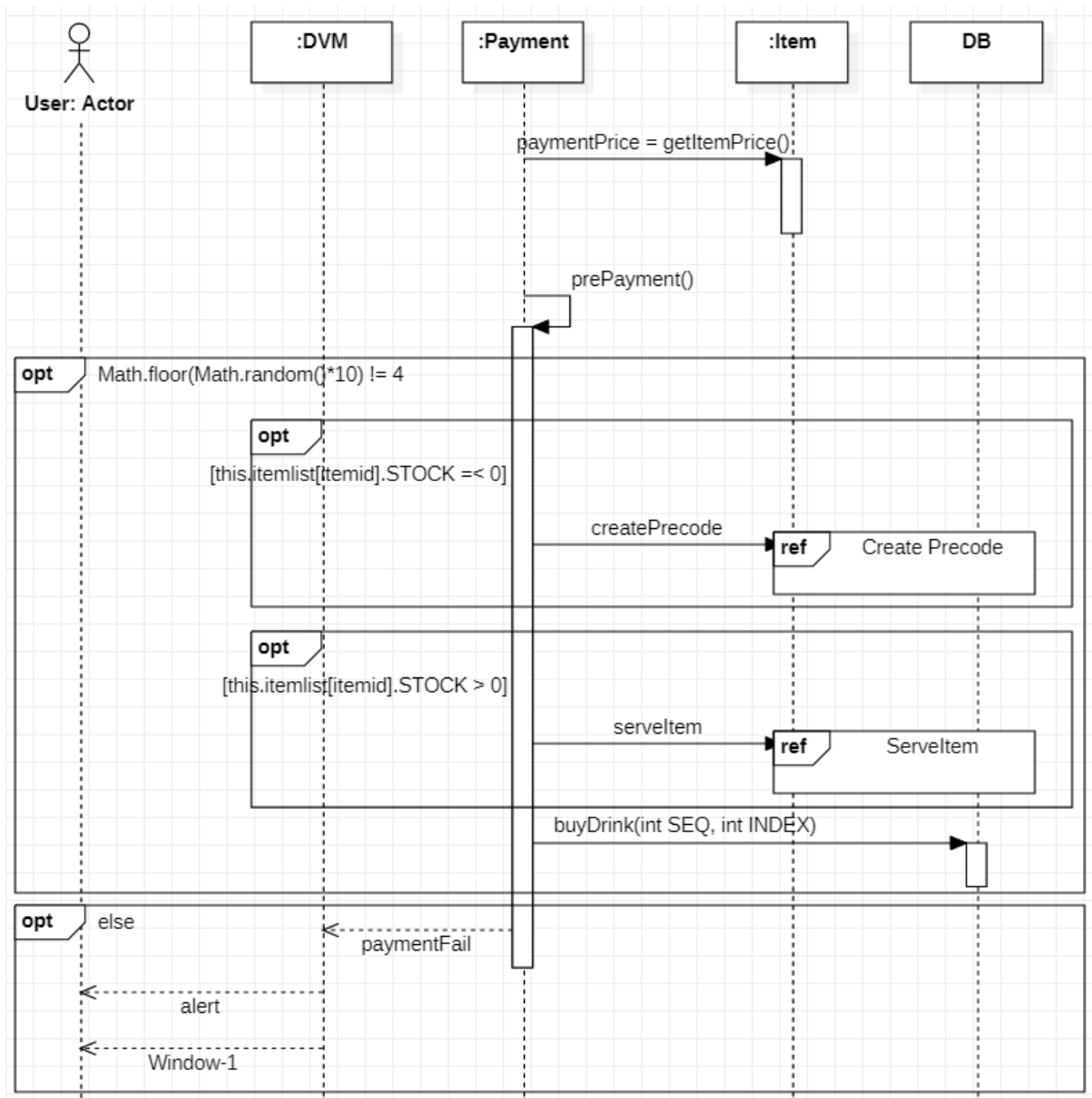


3.1.6. Determine Prepayment

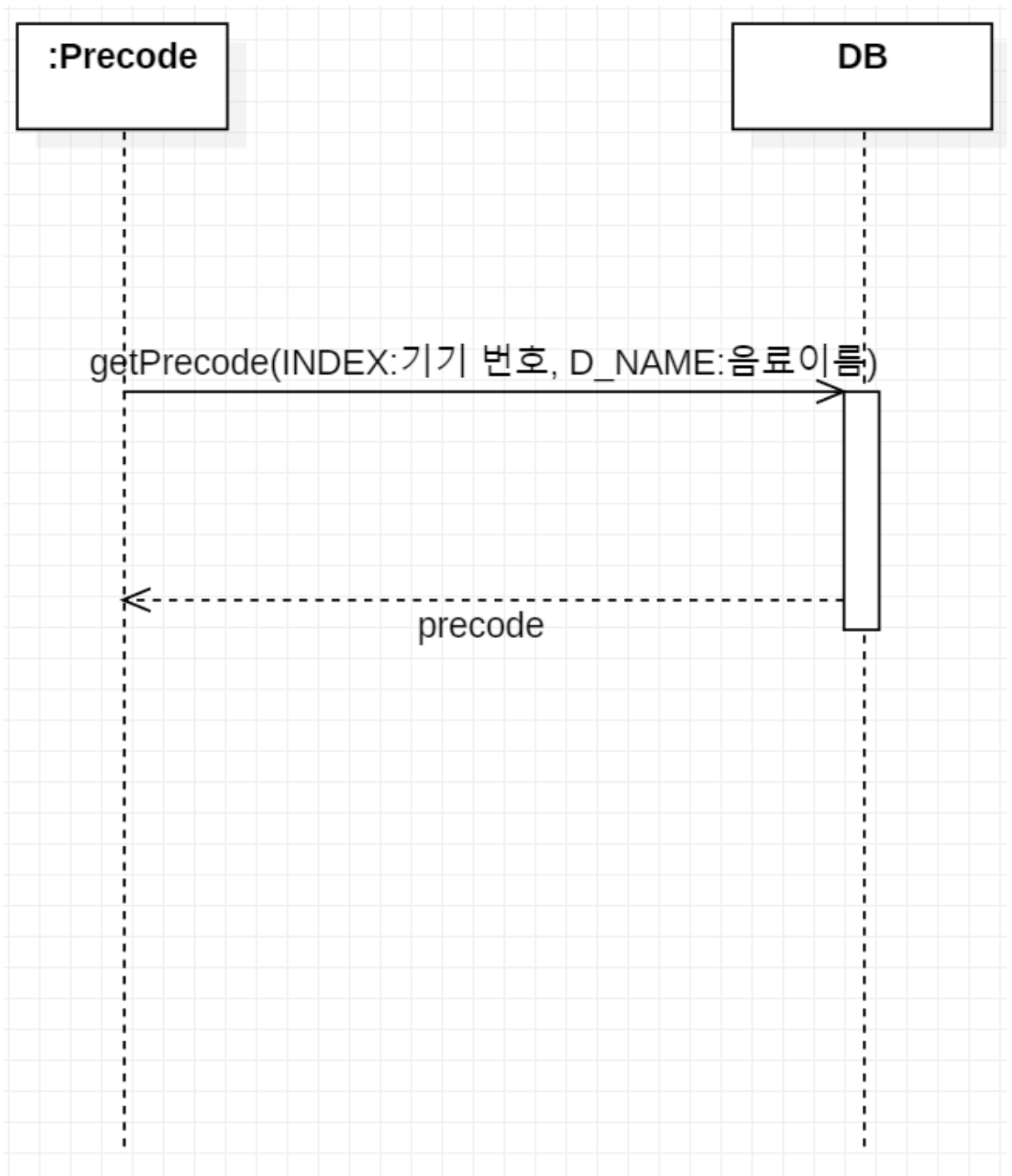


1.

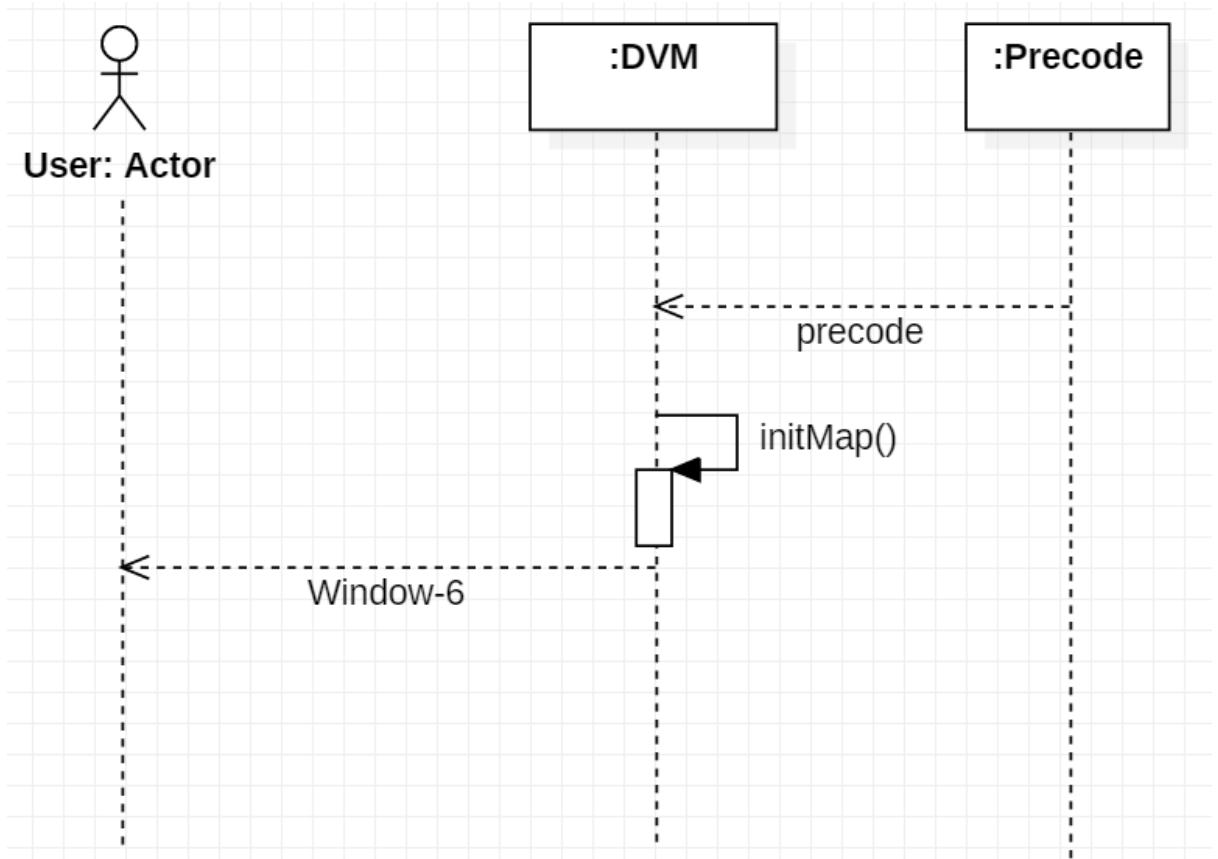
3.1.7. Payment



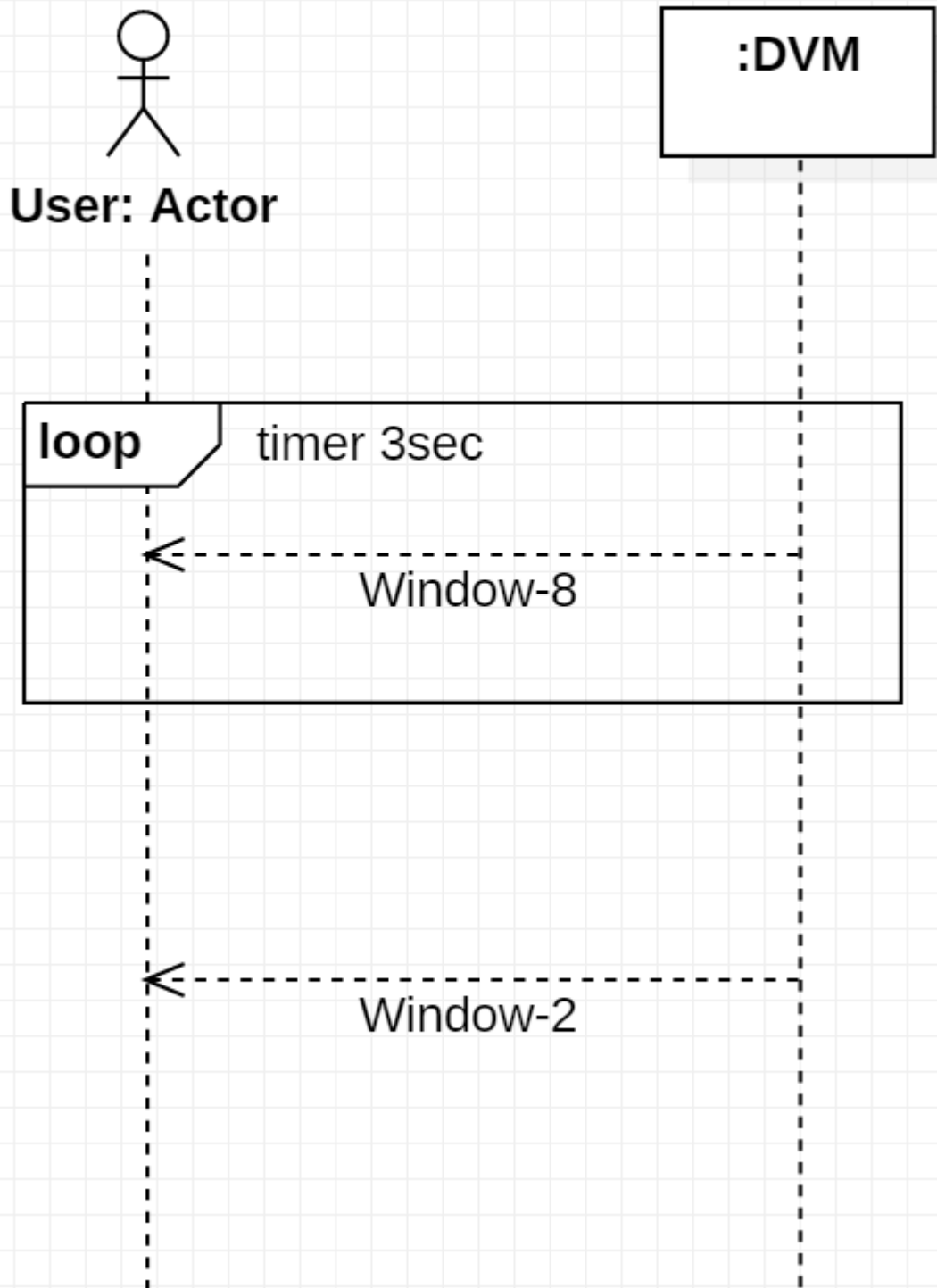
3.1.8. Create Precode



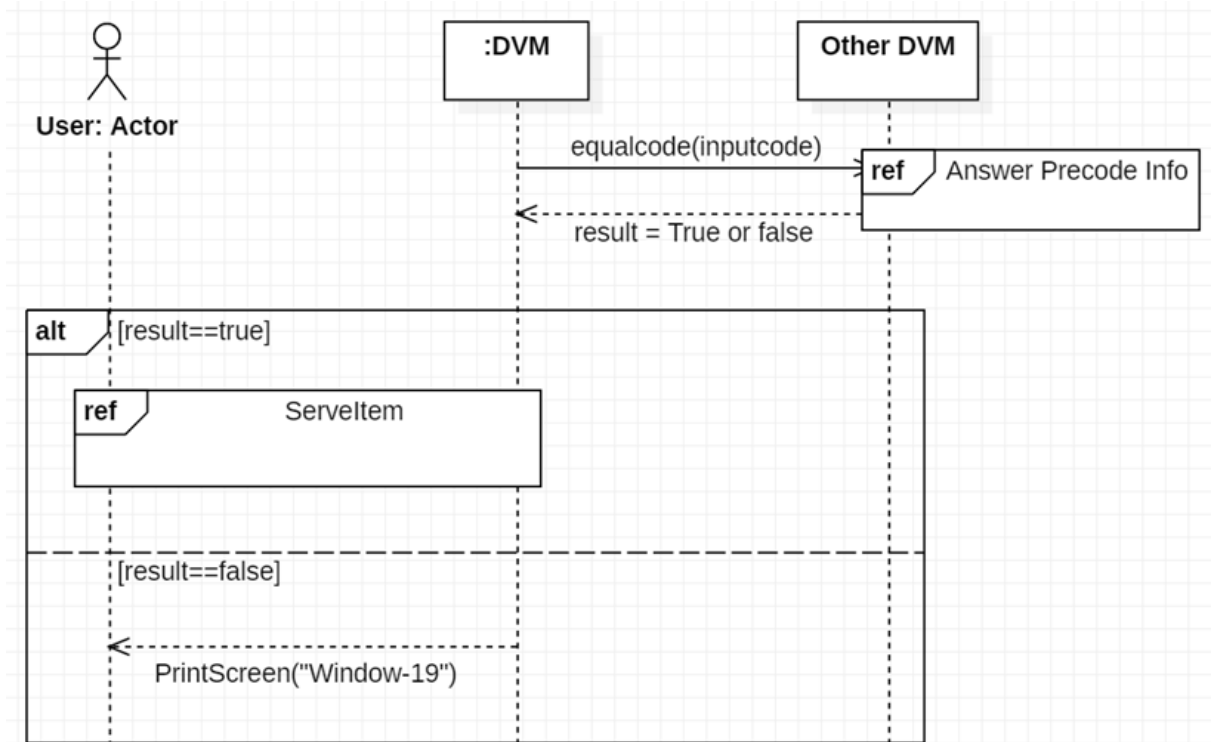
3.1.9. Complete Prepayment



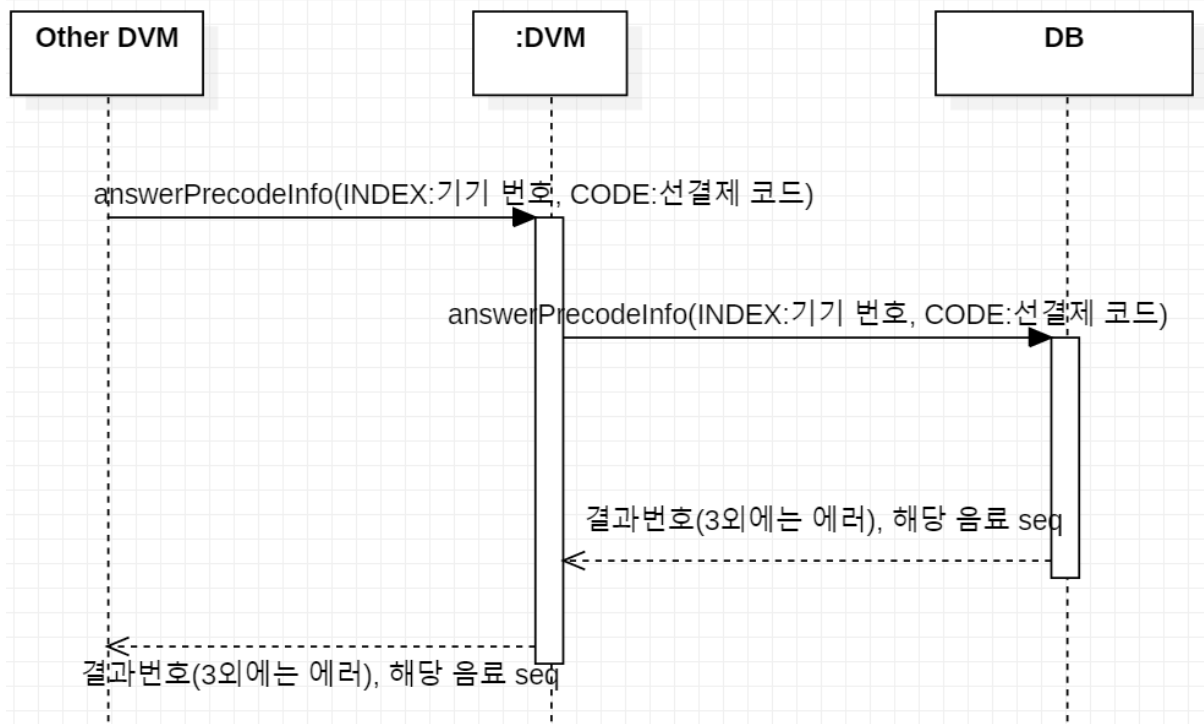
3.1.10. Serve Item



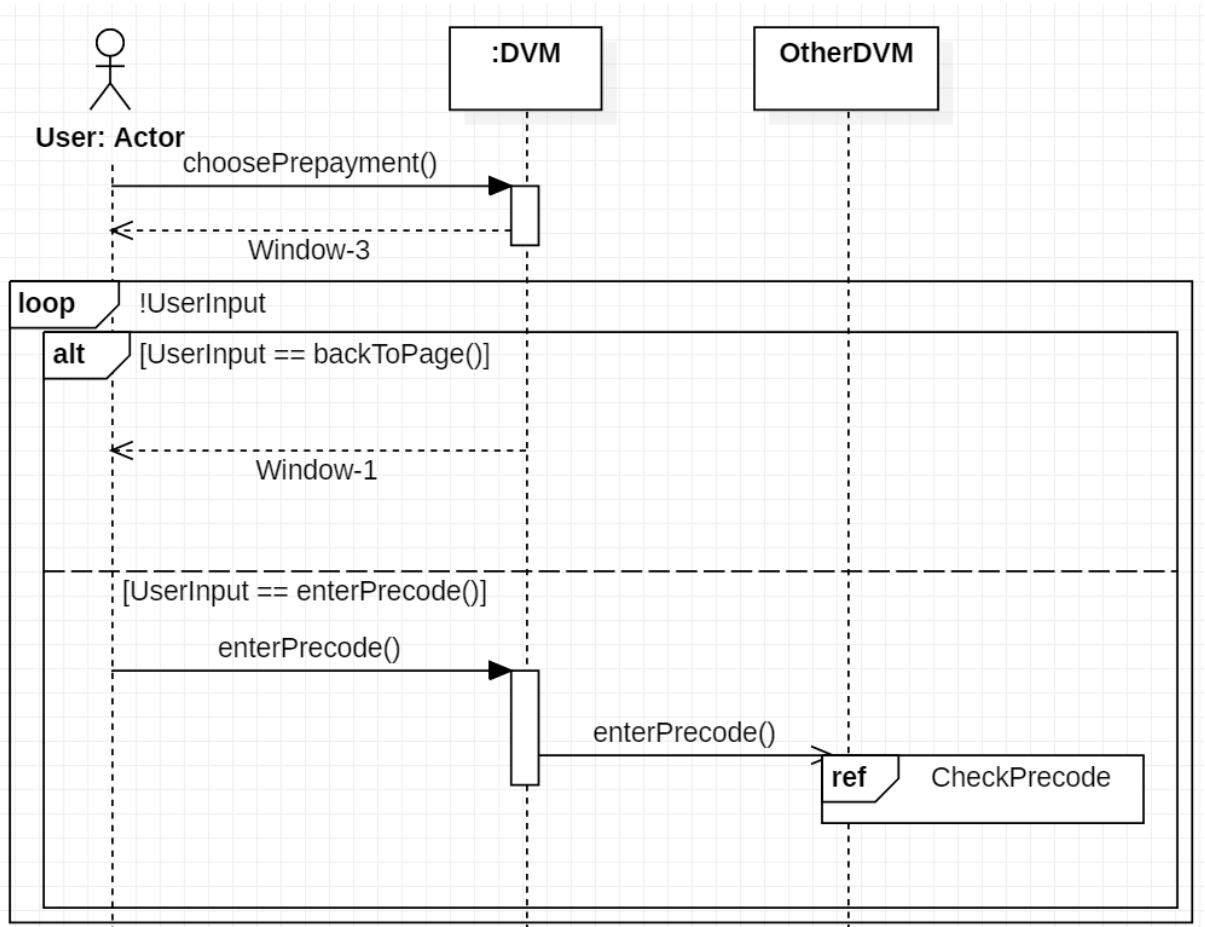
3.1.11. Check Precode



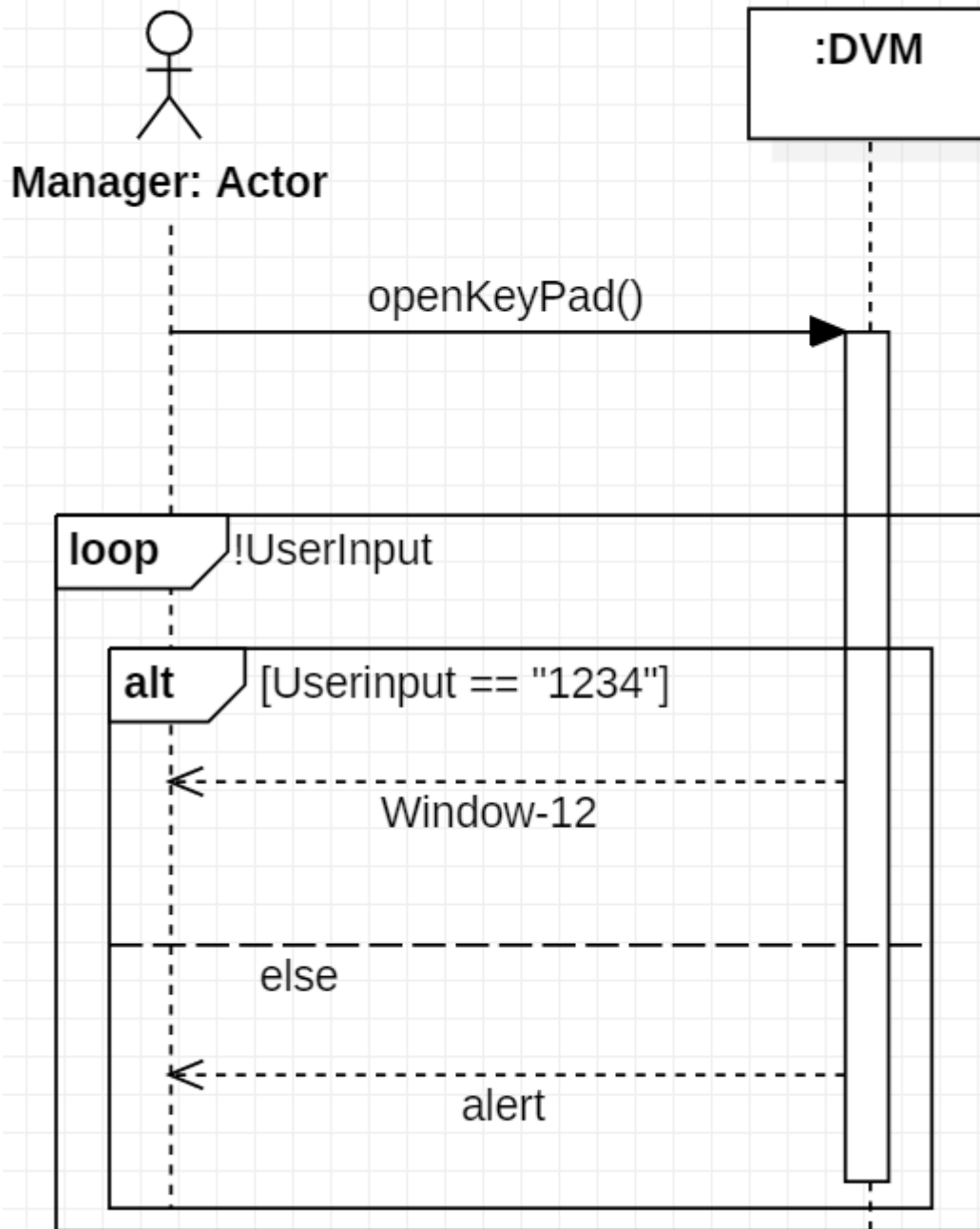
3.1.12. Answer Precode Info



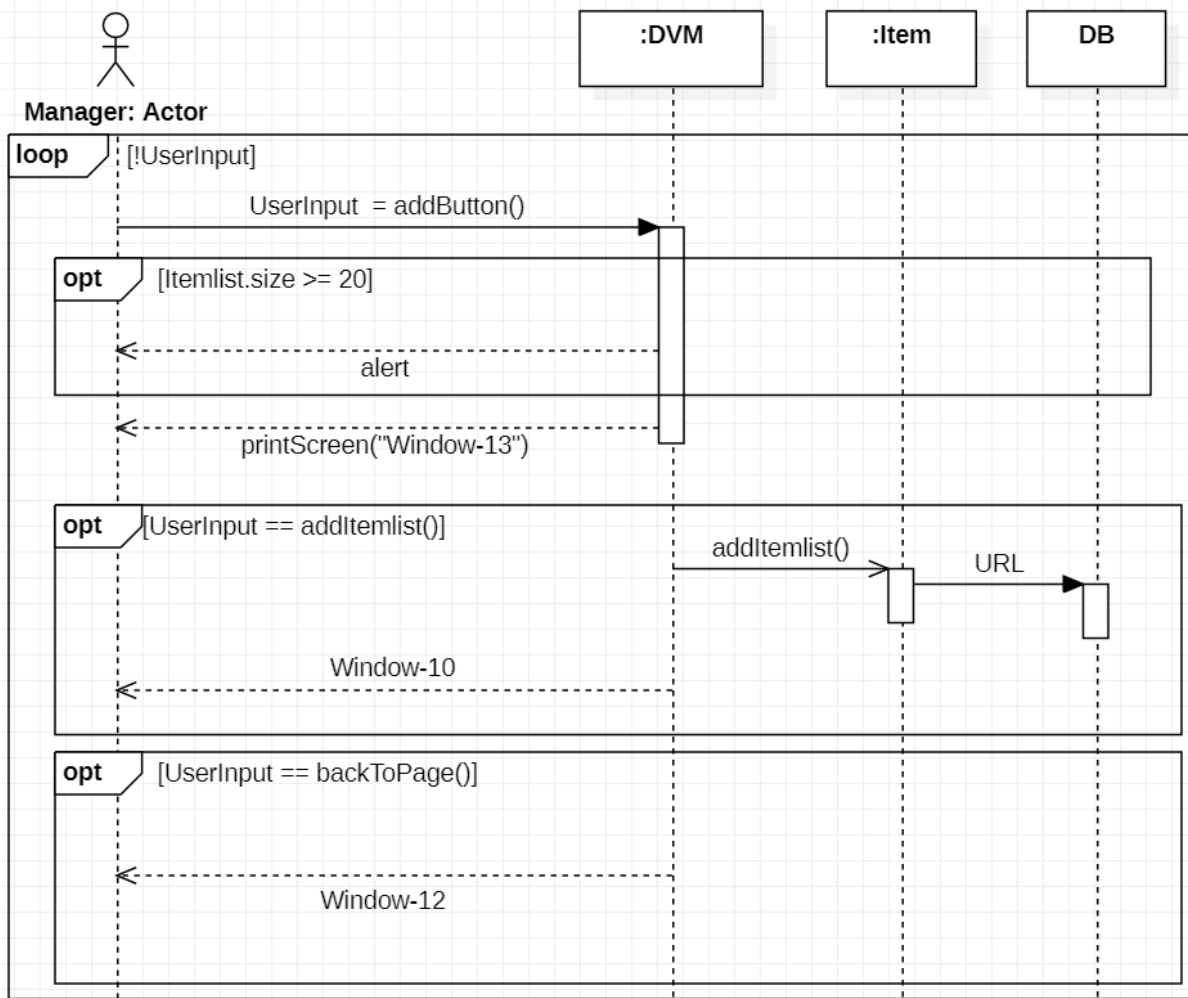
3.1.13. Input Precode



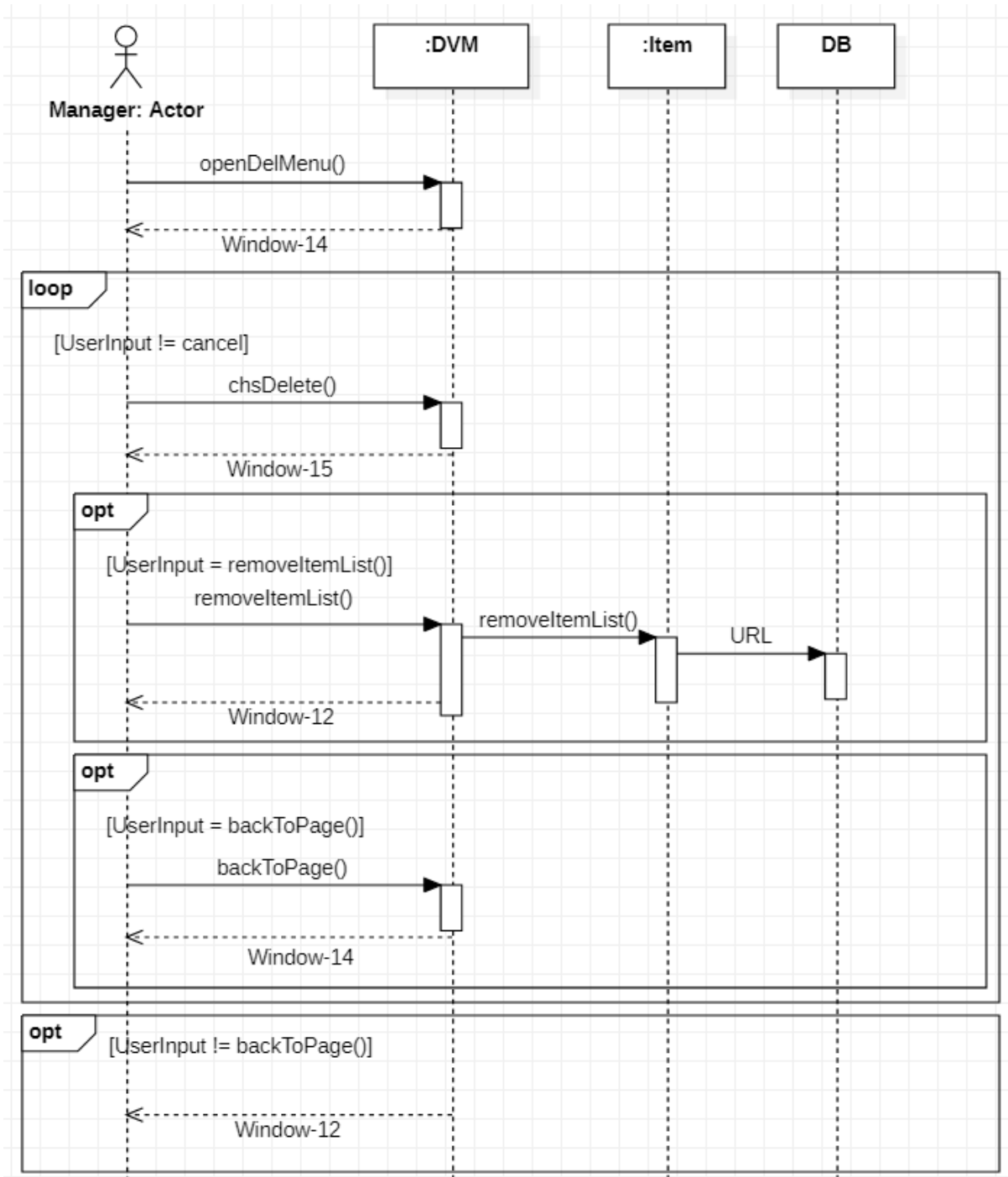
3.1.14. Enter Administrator Mode



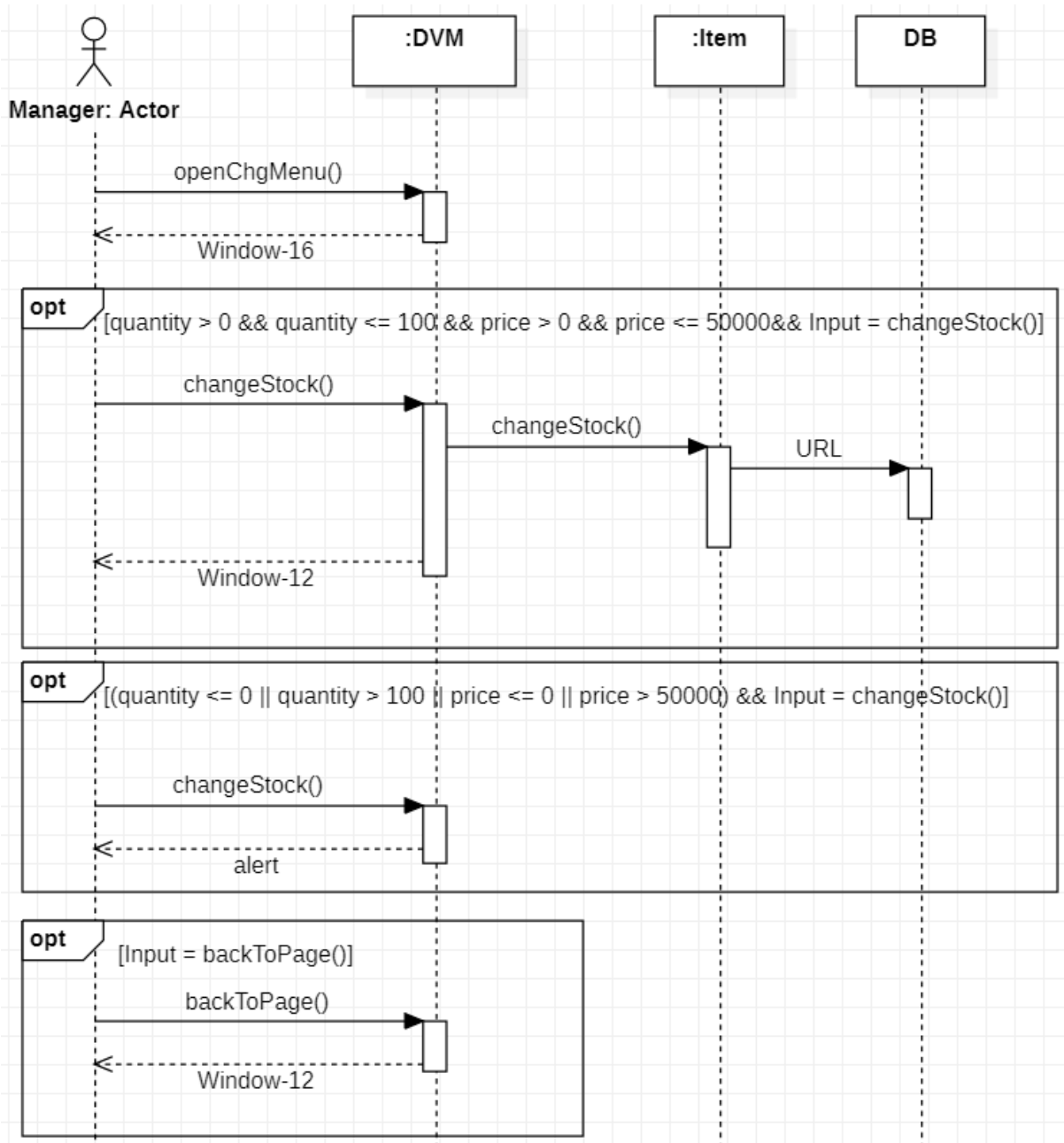
3.1.15. Add Itemlist



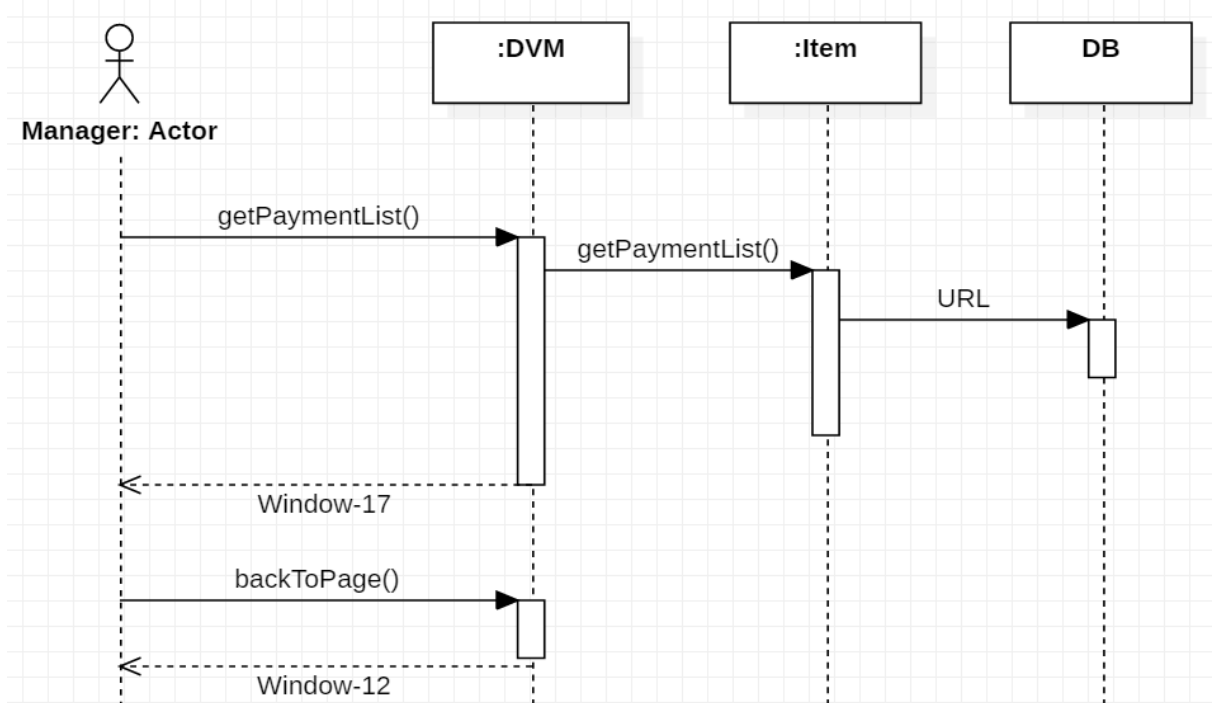
3.1.16. Remove Itemlist



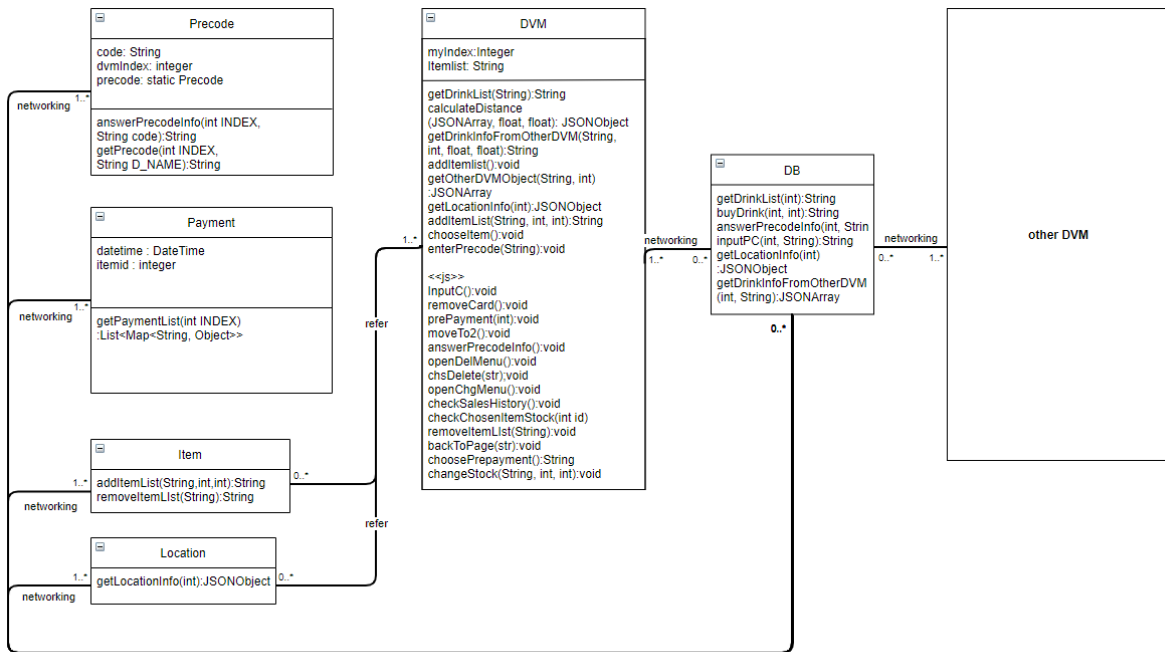
3.1.17. Change Stock&Price



3.1.18. Check Sales History



4. Class Diagram



5. Traceability Analysis

System Function	Essential Use Case	System Operation	Method	class
start order	Start Order	inputC()	getDrinkList(String):String	DVM
input choice	Input Choice	chooseItem()	calculateDistance (JSONArray, float, float): JSONObject	
check chosen item stock	Check Chosen Item Stock	prePayment()	getDrinkInfoFromOtherDVM(String, int, float, float):String	
answer chosen item stock	Answer Chosen Stock Info	choosePrepayment()	addItemList():void	
determine location	Determine Location	enterPrecode()	getOtherDVMObject(String, int):JSONArray	
determine prepayment	Determine Prepayment	openKeyPad()	getLocationInfo(int):JSONObject	
payment	Payment	addButton()	addItemList(String, int, int):String	
create precode	Create Precode	addItemList()	inputC():void	
complete prepayment	Complete Prepayment	openDelMenu()	removeCard():void	
serve item	Serve Item	chsDelete()	prePayment(int):void	
check precode	Check Precode	removeItemList()	moveTo2():void	
answer precode info	Answer Precode Info	openChgMenu()	answerPrecodeInfo():void	
input precode	Input Precode	changeStock()	openDelMenu():void	
enter administrator mode	Enter Administrator mode	getSellRecord()	chsDelete(str):void	
add itemlist	Add Itemlist		openChgMenu():void	
remove itemlist	Remove Itemlist		checkSalesHistory():void	
change stock	Change Stock		checkChosenItemStock(int id)	
change item price	Change Item Price		removeItemList(String):void	
check sales History	Check Sales History		backToPage(str):void	
			choosePrepayment():String	
			changeStock(String, int, int):void	
			answerPrecodeInfo(int INDEX, String code):String	Precode
			getPrecode(int INDEX, String D_NAME):String	
			getPaymentList(int INDEX) :List<Map<String, Object>>	Payment
			addItemList(String,int,int):String	Item
			removeItemList(String):String	
			getLocationInfo(int):JSONObject	Location