



DVM OOPT 2040

객체지향개발방법론 3팀

권민혁, 노성준, 이서준

목차

2141

Design Real Use Case

2142

Define Reports, UI,
and Storyboards

2143

Refine System
Architecture

2144

Define Interaction
Diagrams

2145

Define Design Class
Diagrams

2146

Design Traceability
Analysis

2141. Design Real Use Case

Use Case	UC-01. 음료 선택
Actor	User
Purpose	사용자가 원하는 음료를 선택한다.
Overview	사용자가 화면에 표시된 메뉴에서 음료 ID를 입력하고, 유효성 확인을 수행한다.
Type	Primary
Cross Reference	FR-01, FR-02
Pre-Requisites	사용자에게 DVM의 모든 메뉴를 출력해준다.
Typical Course of Events	(U) : User, (D) : DVM 1. (U) 사용자는 원하는 음료의 [ID, 원하는 수량]을 입력한다. 2. (D) DVM은 입력한 음료 ID와 수량이 유효한지 확인한다. 3. (D) DVM은 사용자가 입력한 음료 ID에 해당하는 음료의 재고가 사용자로부터 받은 수량보다 크거나 같은지 확인한다. 4. (D) 음료의 재고가 있다면 사용자에게 결제 요청 화면을 보여준다.
Alternative Courses of Events	1. (D) 음료의 재고가 없다면 다른 모든 DVM에게 해당 음료의 재고를 요청하여 확인한다. 2. (D) DVM은 자신과 가장 가깝고, 사용자가 구매하고자 하는 음료의 재고가 있는 자판기의 위치를 계산한 후, 선결제 요청 화면을 보여준다.
Exceptional Courses of Events	E1. 잘못된 입력(음료 ID, 재고)을 받은 경우 에러 메시지를 출력하고, UC-01로 돌아간다.

2141. Design Real Use Case

Use Case	UC-02. 결제 요청
Actor	User
Purpose	사용자가 선택한 음료를 결제한다.
Overview	사용자가 카드 정보를 입력하면 DVM은 카드 번호와 잔액을 검증하고, 결제 처리를 수행한다.
Type	Primary
Cross Reference	FR-03, FR-05
Pre-Requisites	UC-01에서 음료 재고가 있음
Typical Course of Events	(U) : User, (D) : DVM, (B) : Bank 1. (U) 사용자는 카드 번호를 입력한다. 2. (D) DVM은 사용자가 입력한 카드 번호로 Bank 에게 카드를 요청한다. 3. (B) Bank는 카드 번호가 유효하다면,카드 번호에 해당하는 카드를 반환한다. 4. (D) DVM은 카드의 잔액이 음료 구매가보다 크거나 같을 경우,음료의 재고를 반환하고 카드의 잔액을 음료 구매가 만큼 차감한다. 5. (D) DVM은 사용자에게 음료를 제공한다.
Alternative Courses of Events	N/A
Exceptional Courses of Events	E1. 잔액이 부족한 경우 에러 메시지를 출력한 후 UC-01로 돌아간다. E2. 잘못된 카드 번호를 입력했을 시 에러 메시지를 출력하고 카드 번호를 다시 입력 받는다(최대 3회). 3번 실패하면 UC-01로 돌아간다.

2141. Design Real Use Case

Use Case	UC-03. 선결제 요청
Actor	User
Purpose	사용자는 다른 DVM에 존재하는 음료를 선결제 한다.
Overview	사용자가 주문한 음료의 재고가 없는 경우 재고가 있는 자판기에 선결제 요청을 보낸다. 응답 결과에 따라 인증코드와 위치를 사용자에게 제공하거나 실패 메시지를 출력한다.
Type	Primary
Cross Reference	FR-04, FR-05
Pre-Requisites	UC-01에서 음료 재고가 없음
Typical Course of Events	(U) : User, (D) : DVM, (B) : Bank 1. (U) 사용자는 선결제 의사를 입력한다. 2. (U) 선결제하고자 하는 사용자는 카드 번호를 입력한다. 3. (D) DVM은 사용자가 입력한 카드 번호로 Bank 에게 카드를 요청한다. 4. (B) Bank는 카드 번호가 유효하다면,카드 번호에 해당하는 카드를 반환한다. 5. (D) DVM은 카드의 잔액이 음료 구매가보다 크거나 같을 경우,인증코드를 생성한다. 6. (D) DVM은 사용자가 선결제하고자 하는 다른 DVM에게 생성한 인증코드와 선결제하고자 하는 음료 id,수량정보를 포함해서 선결제 요청을 보낸다. 7. (D) DVM은 다른 DVM에게 선결제 성공 응답을 받으면,사용자에게 인증코드를 제공한다.
Alternative Courses of Events	사용자가 선결제 의사가 없을 시 UC-01로 돌아간다.
Exceptional Courses of Events	E1. 사용자가 10초 동안 선결제 여부를 입력하지 않는 경우 UC-01로 돌아간다. E2. 잘못된 카드 번호를 입력했을 시 에러 메시지를 출력하고 카드 번호를 다시 입력 받는다(최대 3회). 3번 실패하면 UC-01로 돌아간다. E3. 잔액이 부족한 경우 에러 메시지를 출력한 후 UC-01로 돌아간다. E4. 선결제 요청을 보낸 다른 DVM에 음료의 재고가 없는 경우 에러 메시지를 출력한 후 UC-01로 돌아간다. E5. 다른 DVM과의 통신이 10ms이내로 오지 않으면 에러 메시지를 출력한 후 UC-01로 돌아간다.

2141. Design Real Use Case

Use Case	UC-04. 인증 코드 입력 후 음료 수령
Actor	User
Purpose	사용자는 인증코드를 입력한 후 선결제한 음료를 수령한다.
Overview	사용자가 인증코드를 입력하면, 유효성 확인 후 음료를 제공한다.
Type	Primary
Cross Reference	FR-06
Pre-Requisites	UC-03 완료, 인증 코드 수령
Typical Course of Events	(U) : User, (D) : DVM 1. (U) 사용자는 인증 코드를 입력한다. 2. (D) DVM은 사용자가 입력한 인증코드가 유효한지 검사한다. 3. (D) 인증코드가 유효하다면 DVM은 사용자에게 음료를 제공한다.
Alternative Courses of Events	N/A
Exceptional Courses of Events	E1. 잘못된 인증코드를 입력했을 시 에러 메시지를 출력하고 인증코드를 다시 입력 받는다(최대 3회). 3번 실패하면 UC-01로 돌아간다.

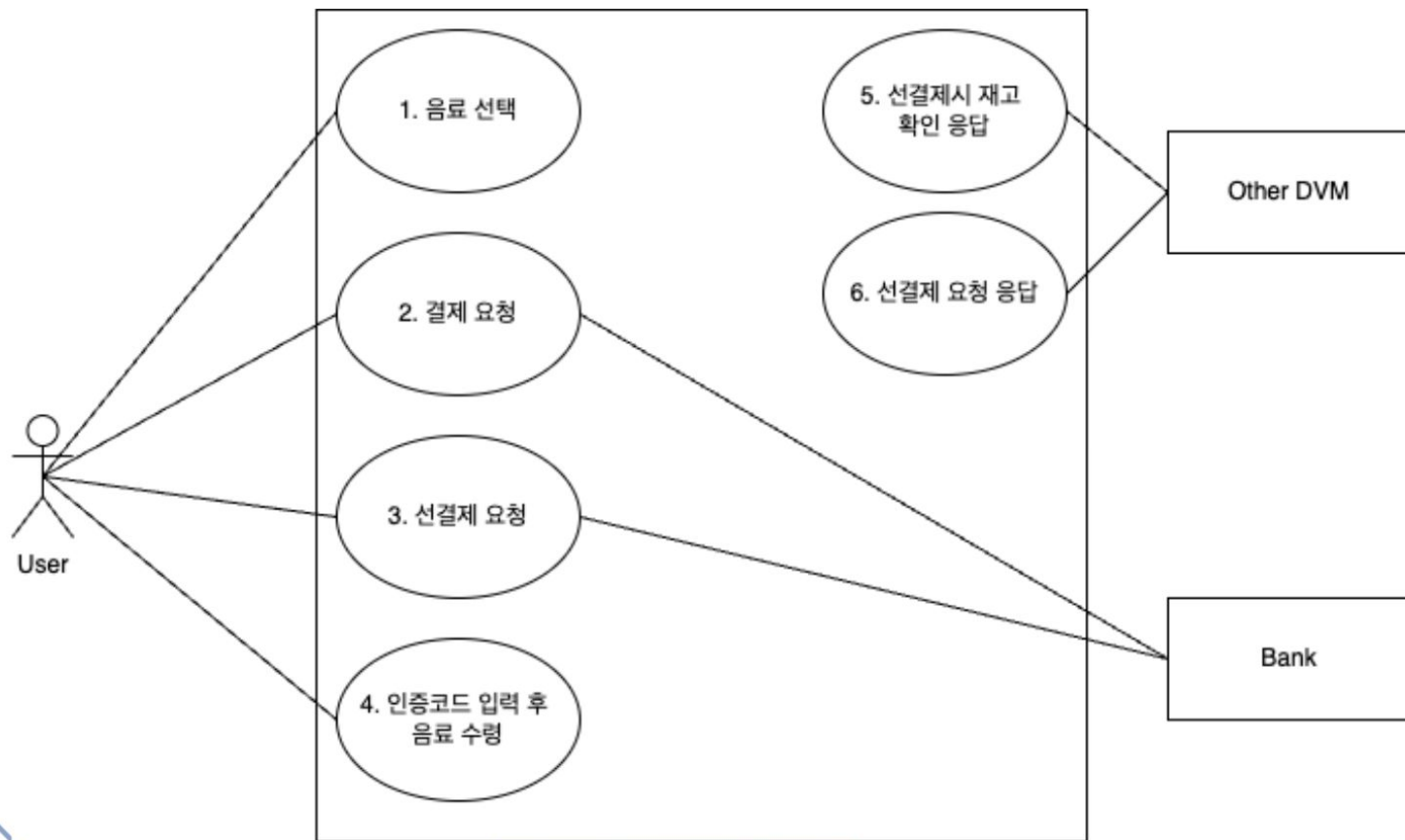
2141. Design Real Use Case

Use Case	UC-05. 선결제 시 재고 확인 응답
Actor	Other DVM
Purpose	DVM은 다른 DVM의 재고 확인 요청에 대해 응답한다.
Overview	DVM은 재고 확인 요청에 해당하는 음료의 재고과 자신의 위치를 요청을 보낸 다른 DVM에 응답한다.
Type	Primary
Cross Reference	FR-07
Pre-Requisites	사용자가 다른 DVM에서 선결제 진행 중이다
Typical Course of Events	(O) : Other DVM (D) : DVM 1. (O) 다른 DVM은 DVM에게 특정 음료에 대한 재고 확인 요청을 전송한다. 2. (D) DVM은 다른 DVM으로부터 재고 확인 요청 메시지를 수신한다. 3. (D) DVM은 재고 확인 요청에 포함된 음료의 재고를 확인한다. 4. (D) DVM은 음료의 재고와 자신의 위치 정보를 응답한다.
Alternative Courses of Events	N/A
Exceptional Courses of Events	N/A

2141. Design Real Use Case

Use Case	UC-06. 선결제 요청 응답
Actor	Other DVM
Purpose	DVM은 다른 DVM의 선결제 요청에 대해 응답한다.
Overview	DVM은 선결제 요청에 해당하는 음료의 재고가 있으면 차감하고, 인증코드를 저장한 후 요청한 다른 DVM에 선결제 성공을 응답한다.
Type	Primary
Cross Reference	FR-07
Pre-Requisites	사용자가 다른 DVM에서 선결제 진행 중이다
Typical Course of Events	(O) : Other DVM, (D) : DVM 1. (O) 다른 DVM은 DVM에게 특정 음료에 대한 선결제를 요청한다. 2. (D) DVM은 선결제 요청에 포함된 음료의 재고를 확인한다. 3. (D) DVM은 해당 음료의 재고가 있다면 재고를 차감하고, 선결제 요청에 포함된 인증 코드를 저장한 후, 선결제 성공을 응답한다.
Alternative Courses of Events	N/A
Exceptional Courses of Events	N/A

2141. Design Real Use Case



2142. Define Reports, UI, and Storyboards

UC-01 음료 선택

음료 종류 (음료 코드)			
콜라 (01)	사이다 (02)	녹차 (03)	홍차 (04)
밀크티 (05)	탄산수 (06)	보리차 (07)	캔커피 (08)
물 (09)	에너지링크 (10)	유자차 (11)	식혜 (12)
아이스티 (13)	딸기주스 (14)	오렌지주스 (15)	포도주스 (16)
이온음료 (17)	아메리카노 (18)	핫초코 (19)	카페라떼 (20)

[음료와 수량을 선택해주세요. ex - 3, 2 (녹차 2개)]

입력 : 1, 5

UC-02 결제 요청

[카드 번호를 입력해주세요.]

입력 : 123412341234

[결제가 완료 되었습니다.]

음료 ID : 1

음료명 : 콜라

수량 : 5개

결제 : 성공

2142. Define Reports, UI, and Storyboards

UC-03 선결제 요청

[재고가 존재하는 자판기 위치는 다음과 같습니다.]

x : 5

y : 4

[선결제 하시겠습니까? (Y/F)]

입력 : Y

[카드 번호를 입력해주세요.]

입력 : 123412341234

[인증 코드 발급이 완료되었습니다.]

인증 코드 : 2E5dc

UC-04 인증코드 입력 후 음료 수령

[인증 코드를 입력해주세요.]

입력 : 2E5dc

[인증이 완료 되었습니다.]

음료 ID : 1

음료명 : 콜라

수량 : 5개

2142. Define Reports, UI, and Storyboards

UC-05 선결제시 재고 확인 응답

[1 콜라의 음료 재고를 확인합니다.]

[응답을 기다리는 중...]

[조회 결과]

음료 ID: 1

재고 수량: 5개

자판기 위치: (X: 100, Y: 200)

요청 처리 결과: 성공

UC-06 선결제 요청 응답

[1 콜라의 선결제 요청을 보냅니다.]

[응답을 기다리는 중...]

[선결제 결과]

음료 ID: 1

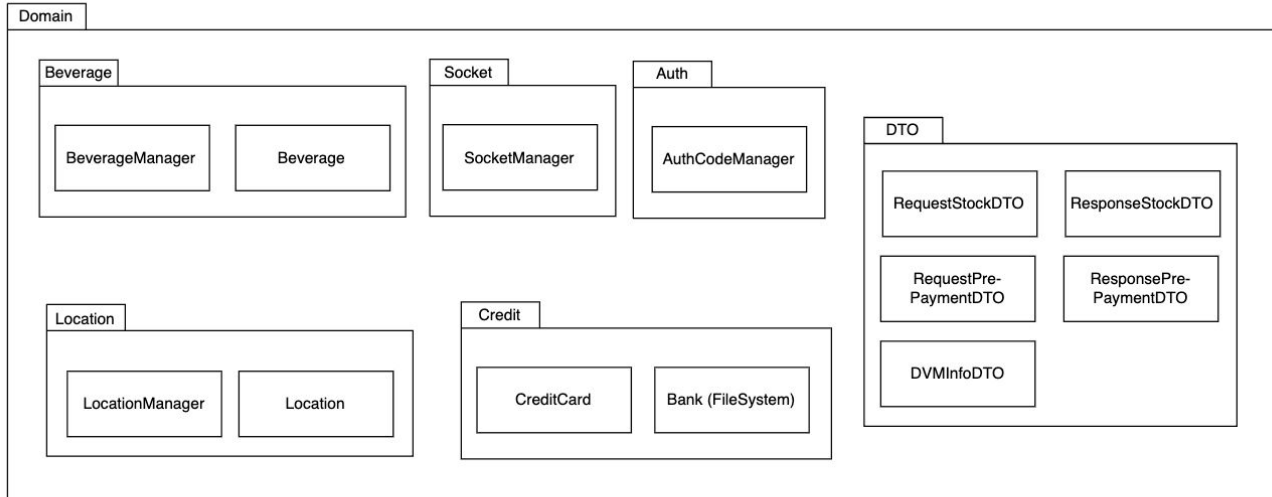
선결제 수량: 2개

인증코드: AB123

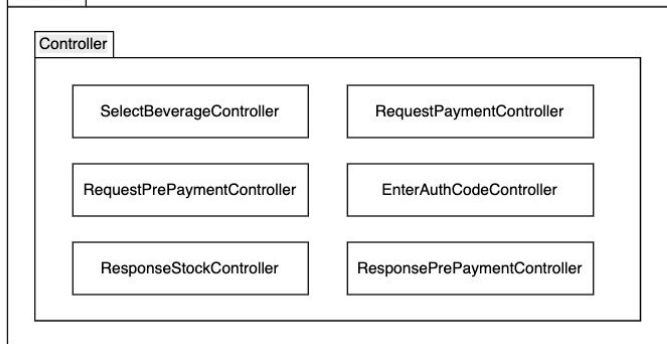
요청 처리 결과: 성공

2143. Refine System Architecture

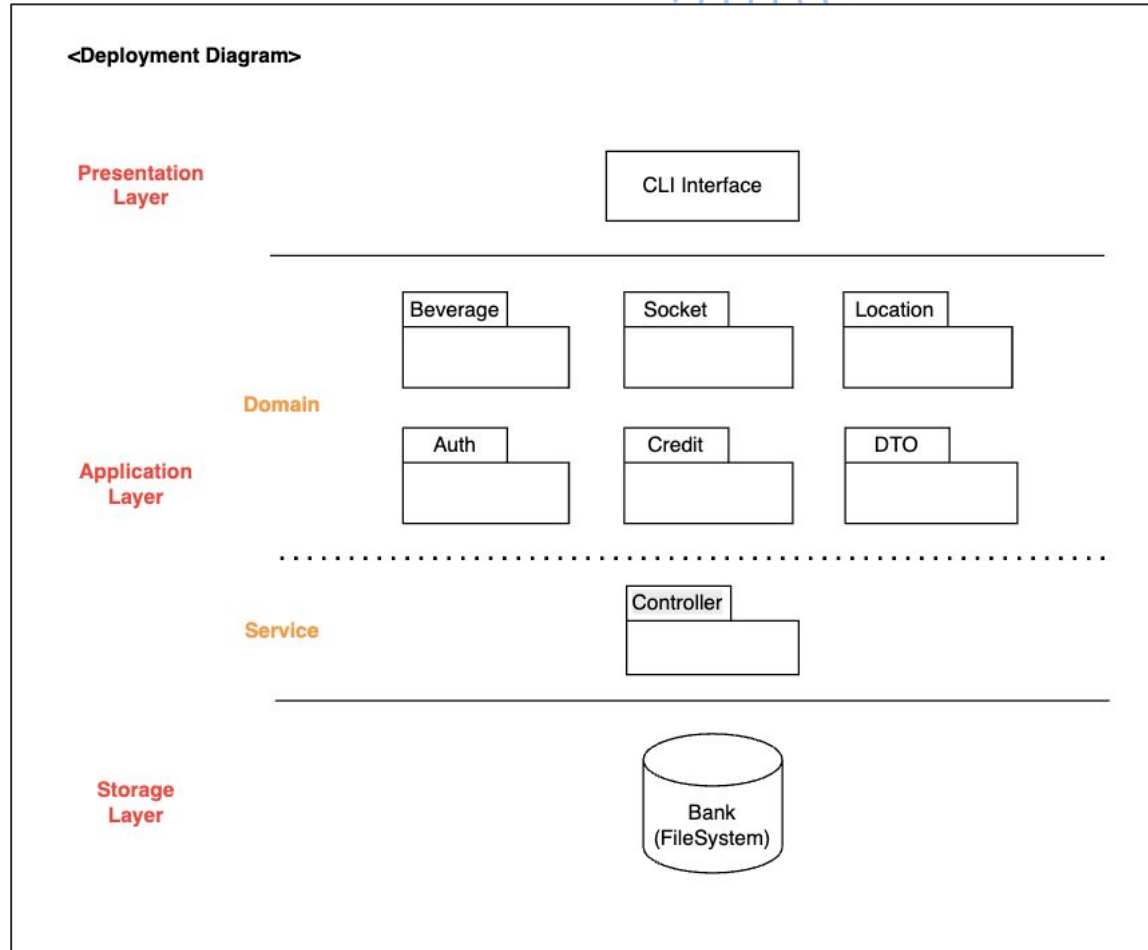
<Package Diagram>



Services



2143. Refine System Architecture



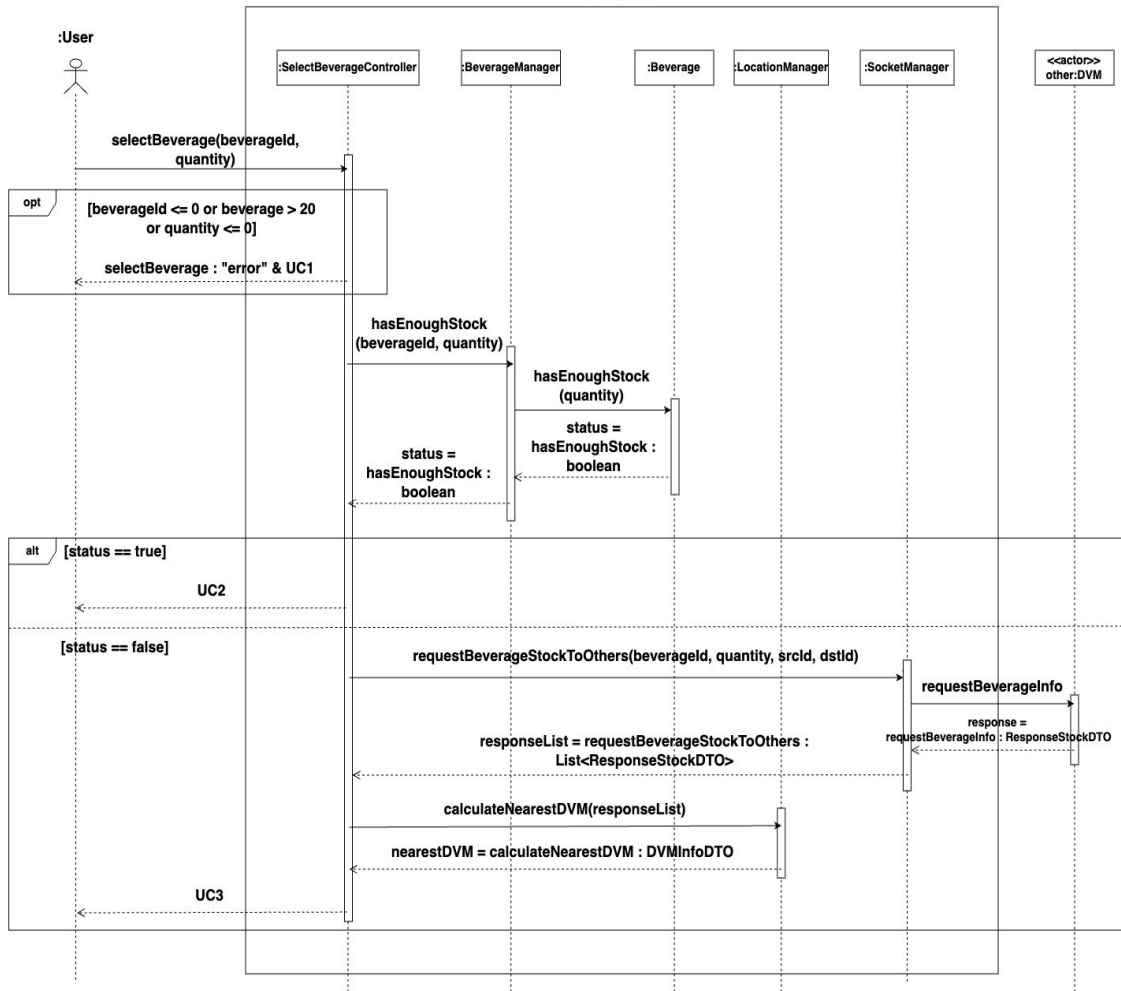
2144. Define Interaction Diagrams

UC1. 음료 선택

1. (U) 사용자는 원하는 음료의 [ID, 원하는 수량]을 입력한다.
2. (D) DVM은 사용자가 입력한 음료 ID가 유효한지 확인한다.
3. (D) DVM은 사용자가 입력한 음료 ID에 해당하는 음료의 재고가 사용자로부터 받은 수량보다 크거나 같은지 확인한다.
4. (D) 음료의 재고가 있다면 사용자에게 결제 요청 화면을 보여준다.
5. (D) 음료의 재고가 없다면 다른 모든 DVM에게 해당 음료의 재고를 요청하여 확인한다.
6. (D) DVM은 자신과 가장 가깝고, 사용자가 구매하고자 하는 음료의 재고가 있는 자판기의 위치를 계산한 후, 선결제 요청 화면을 보여준다.

<Use case 1>

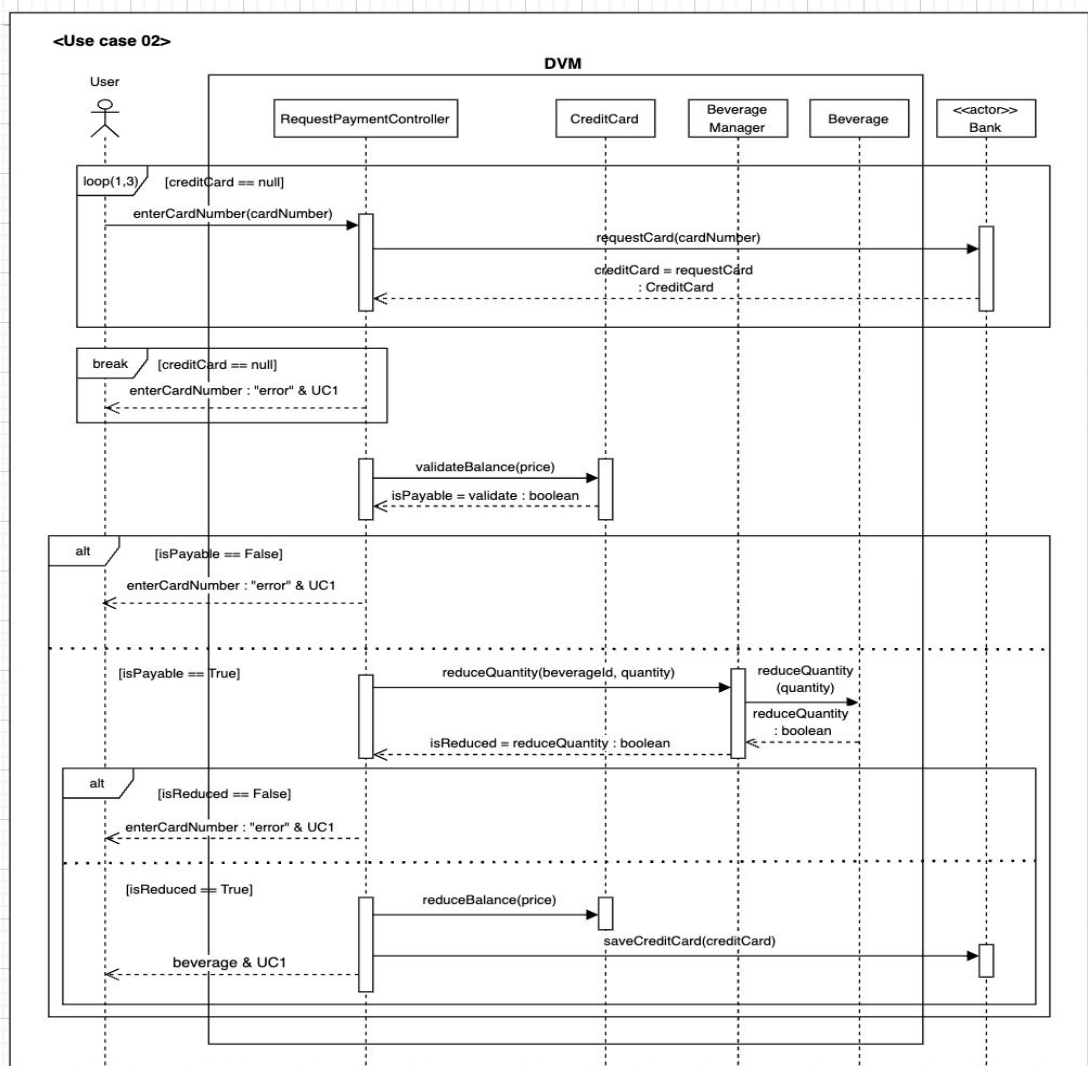
DVM



2144. Define Interaction Diagrams

UC2. 결제 요청

1. (U) 사용자는 카드 번호를 입력한다.
2. (D) DVM은 사용자가 입력한 카드 번호로 Bank에게 카드를 요청한다.
3. (B) Bank는 카드 번호가 유효하다면, 카드 번호에 해당하는 카드를 반환한다.
4. (D) DVM은 카드의 잔액이 음료 구매가보다 크거나 같을 경우, 음료의 재고를 반환하고 카드의 잔액을 음료 구매가만큼 차감한다.
5. (D) DVM은 사용자에게 음료를 제공한다.



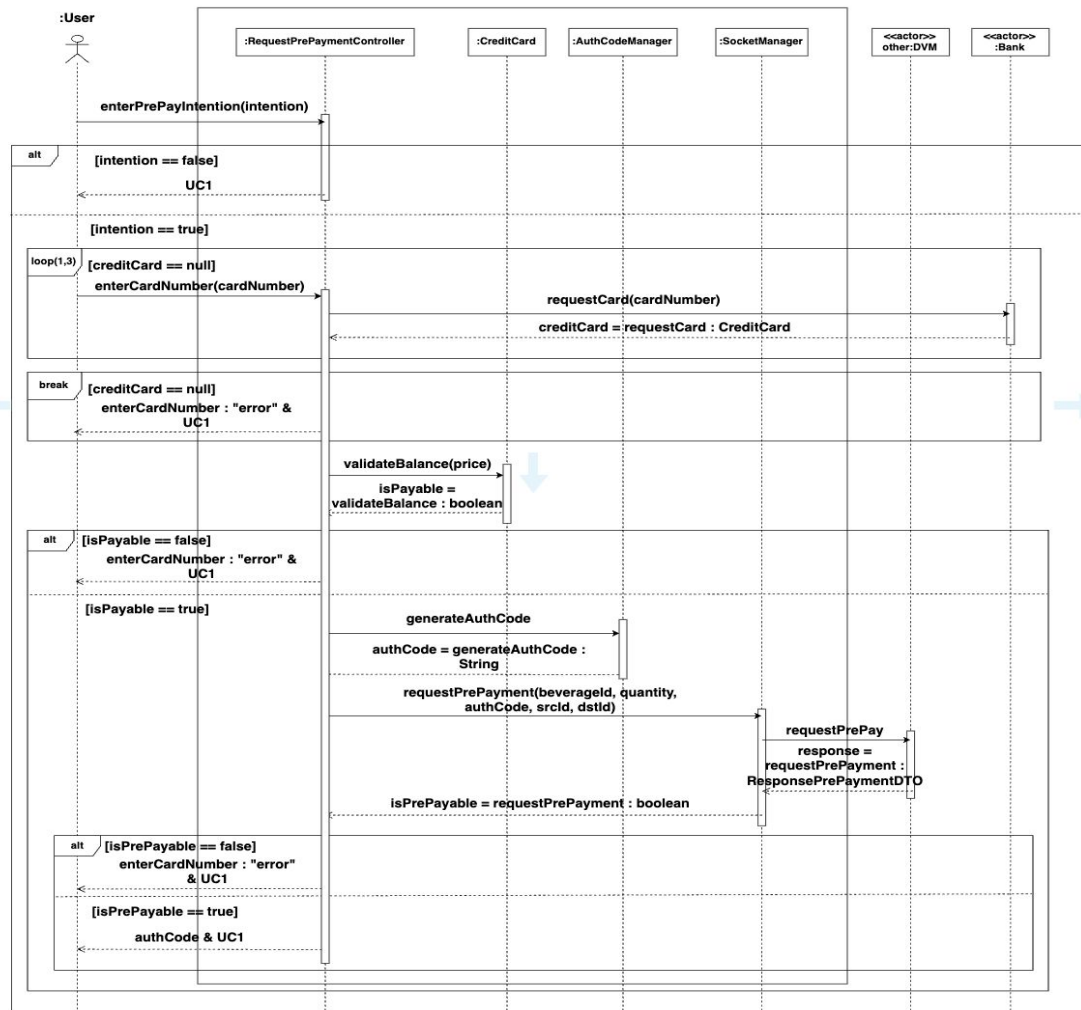
2144. Define Interaction Diagrams

UC3. 선결제 요청

1. (U) 사용자는 선결제 의사를 입력한다.
2. (U) 선결제하고자 하는 사용자는 카드 번호를 입력한다.
3. (D) DVM은 사용자가 입력한 카드 번호로 Bank 에게 카드를 요청한다.
4. (B) Bank는 카드 번호가 유효하다면, 카드 번호에 해당하는 카드를 반환한다.
5. (D) DVM은 카드의 잔액이 음료 구매가보다 크거나 같을 경우, 인증코드를 생성한다.
6. (D) DVM은 사용자가 선결제하고자 하는 다른 DVM에게 생성한 인증코드와 선결제하고자 하는 음료 id, 수량 정보를 포함해서 선결제 요청을 보낸다.
7. (D) DVM은 다른 DVM에게 선결제 성공 응답을 받으면, 사용자에게 인증코드를 제공한다.

<Use case 3>

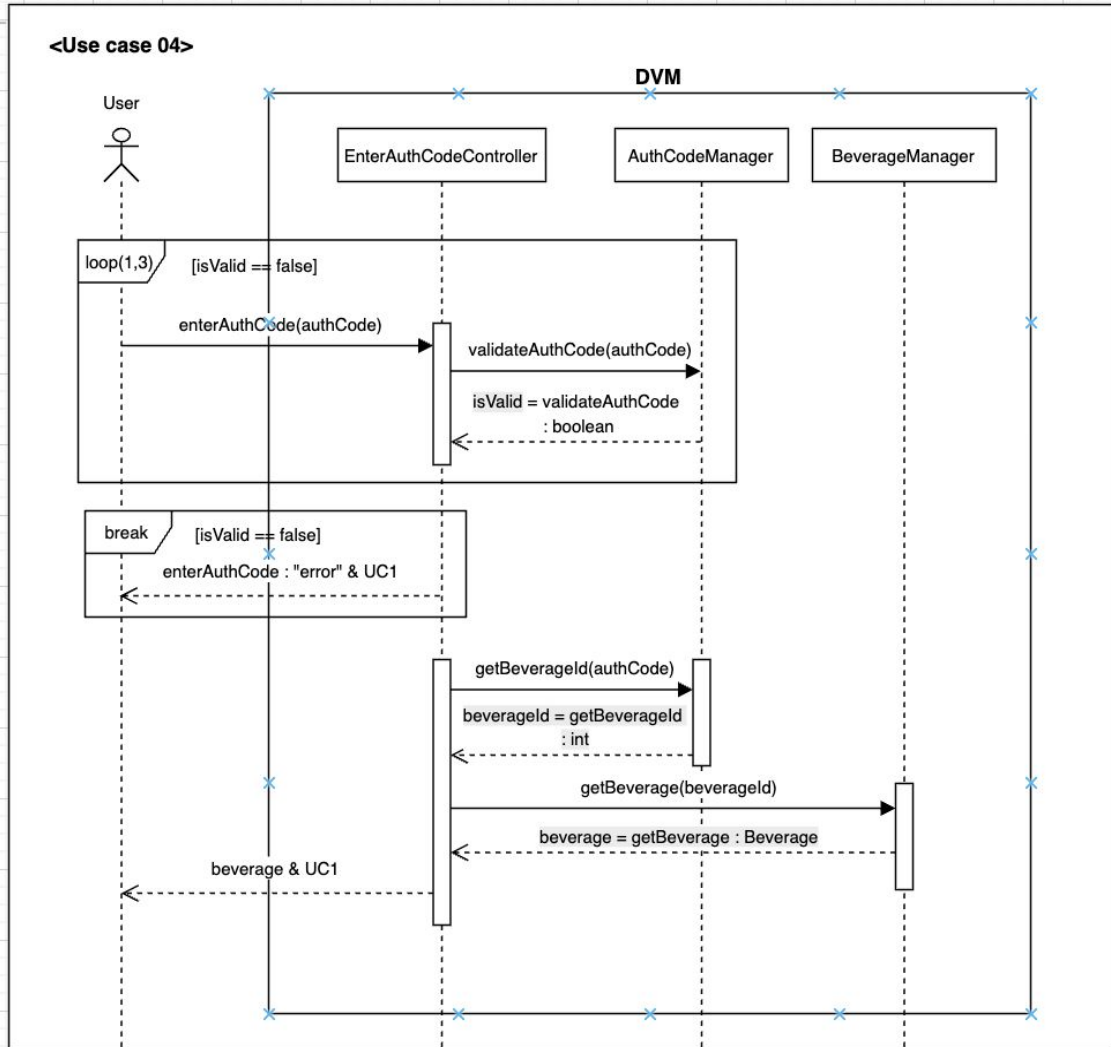
DVM



2144. Define Interaction Diagrams

UC4. 인증 코드 입력 후 음료 수령

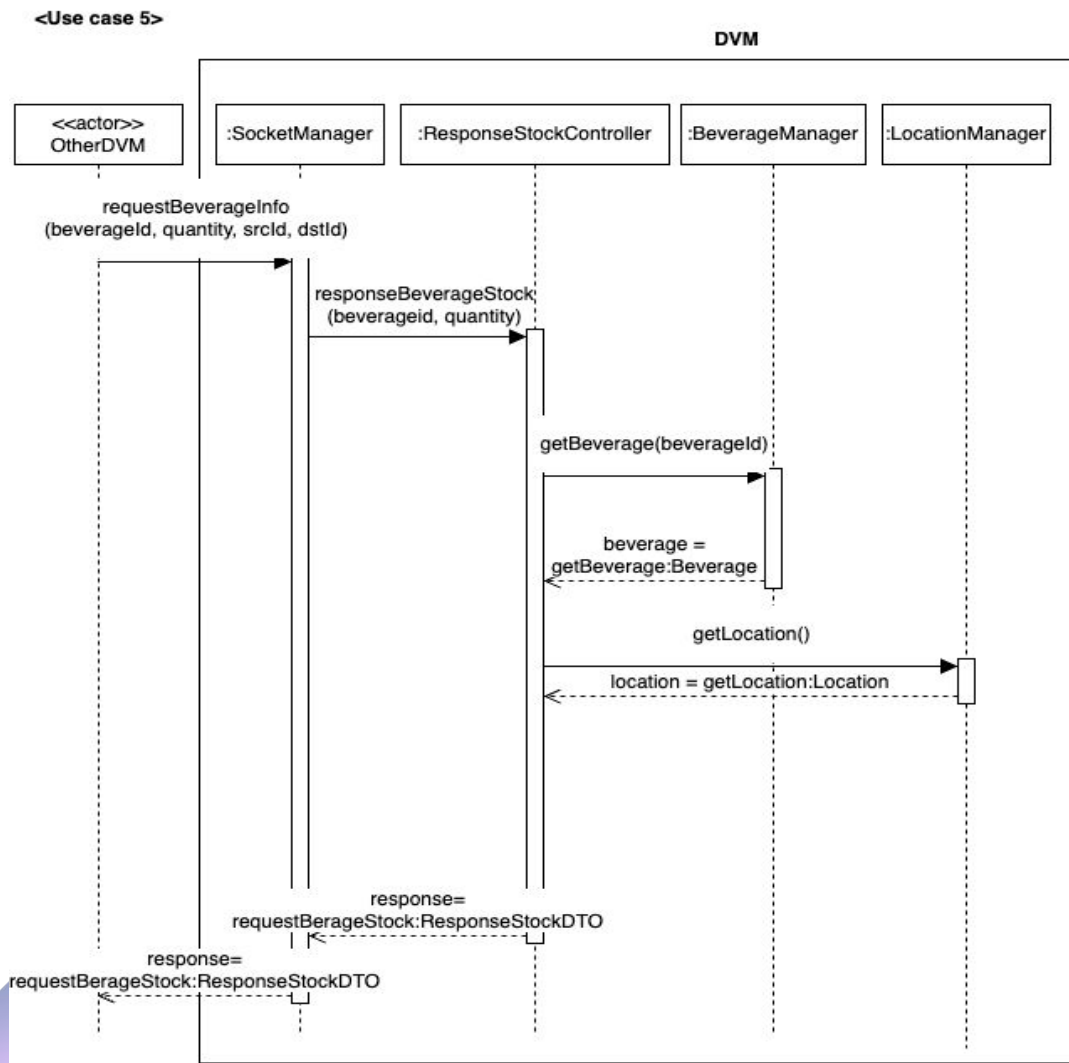
1. (U) 사용자는 인증 코드를 입력한다
2. (D) DVM은 사용자가 입력한 인증코드가 유효한지 검사한다.
3. (D) 인증코드가 유효하다면 DVM은 사용자에게 음료를 제공한다.



2144. Define Interaction Diagrams

UC5. 선결제시 재고 확인 응답

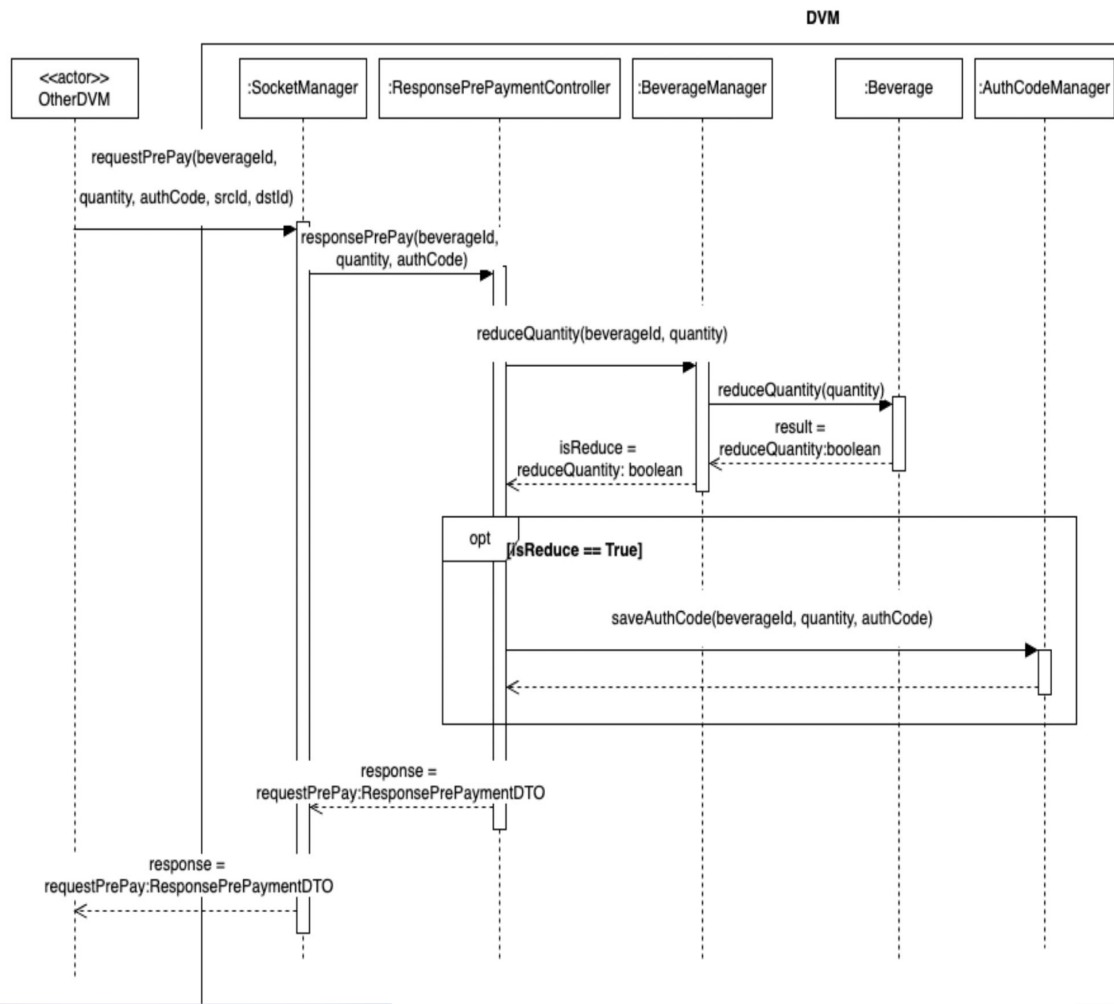
1. (O) 다른 DVM은 DVM에게 특정 음료에 대한 재고 확인 요청을 전송한다.
2. (D) DVM은 다른 DVM으로부터 재고 확인 요청 메시지를 수신한다.
3. (D) DVM은 재고 확인 요청에 포함된 음료의 재고를 확인한다.
4. (D) DVM은 음료의 재고와 자신의 위치 정보를 응답한다.



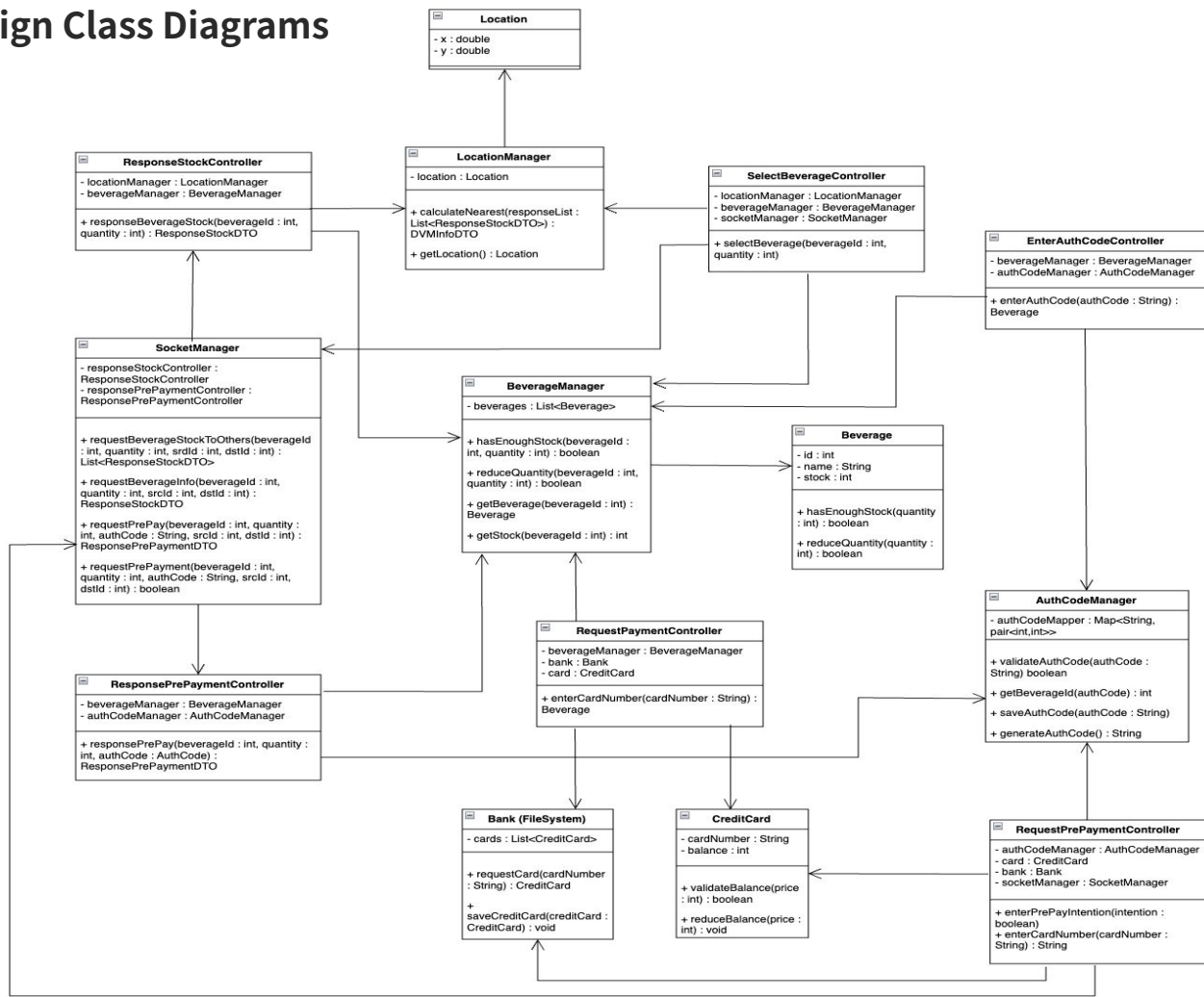
2144. Define Interaction Diagrams

UC6. 선결제 요청 응답

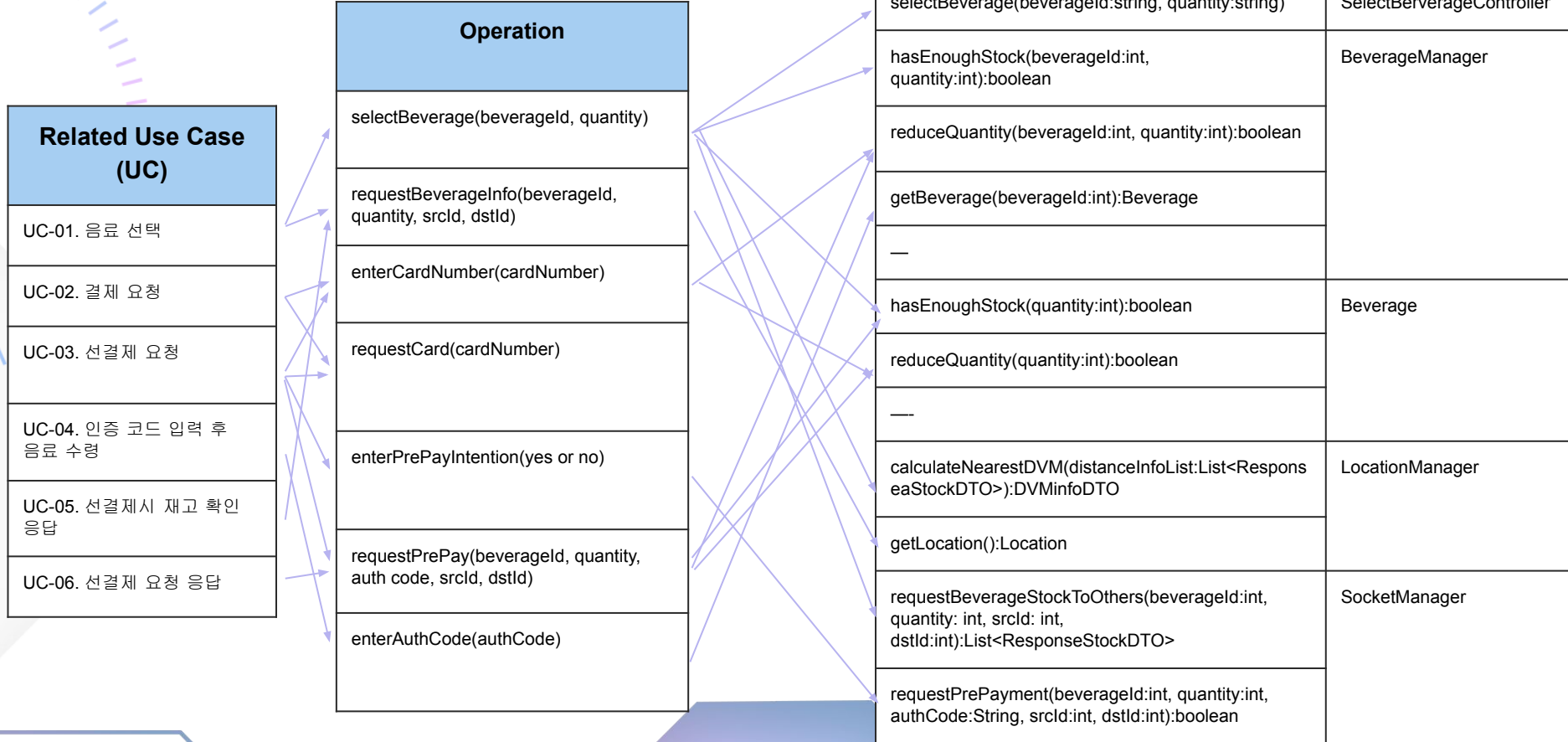
1. (O) 다른 DVM은 DVM에게 특정 음료에 대한 선결제를 요청한다.
2. (D) DVM은 선결제 요청에 포함된 음료의 재고를 확인한다.
3. (D) DVM은 해당 음료의 재고가 있다면 재고를 차감하고, 선결제 요청에 포함된 인증 코드를 저장한 후, 선결제 성공을 응답한다.



2145. Define Design Class Diagrams



2146. Design Traceability Analysis



2146. Design Traceability Analysis

Related Use Case (UC)
UC-01. 음료 선택
UC-02. 결제 요청
UC-03. 선결제 요청
UC-04. 인증 코드 입력 후 음료 수령
UC-05. 선결제시 재고 확인 응답
UC-06. 선결제 요청 응답

Operation
selectBeverage(beverageId, quantity)
requestBeverageInfo(beverageId, quantity, srcId, dstId)
enterCardNumber(cardNumber)
requestCard(cardNumber)
enterPrePayIntention(yes or no)
requestPrePay(beverageId, quantity, auth code, srcId, dstId)
enterAuthCode(authCode)

Method	Class
requestBeverageInfo(beverageId:int, quantity:int, srcId:int, dstId:int):ResponseStockDTO	SocketManager
requestPrePay(beverageId:int, quantity:int, authCode:string, srcId:int, dstId:int):ResponsePrePaymentDTO	
enterCardNumber(cardNumber:String):Beverage	RequestPaymentController
validateBalance(price:int):boolean	CreditCard
reduceBalance(price:int):void	
enterPrePayIntention(Intention:boolean):void	RequestPrePaymentController
generateAuthCode():string	AuthCodeManager
validateAuthCode(authCode:string):boolean	
getBeverageId(authCode:string):int	
saveAuthCode(beverageId:int, quantity:int, authCode:String):void	
requestBeverageInfo(beverageId:int, quantity:int):ResponseStockDTO	ResponseStockController
responsePrePay(beverageId:int, quantity:int, authCode:string):ResponsePrePaymentDTO	ResponsePrePaymentController
requestCard(cardNumber:String):CreditCard	Bank

The background features several decorative geometric elements: a blue-to-purple gradient hexagon in the top-left, a dashed blue arc in the top-right, a dashed purple arc in the middle-left, a light gray hexagon in the bottom-left, a blue-to-purple gradient hexagon in the bottom-center, and a light gray hexagon with blue and purple outlines in the bottom-right.

감사합니다.