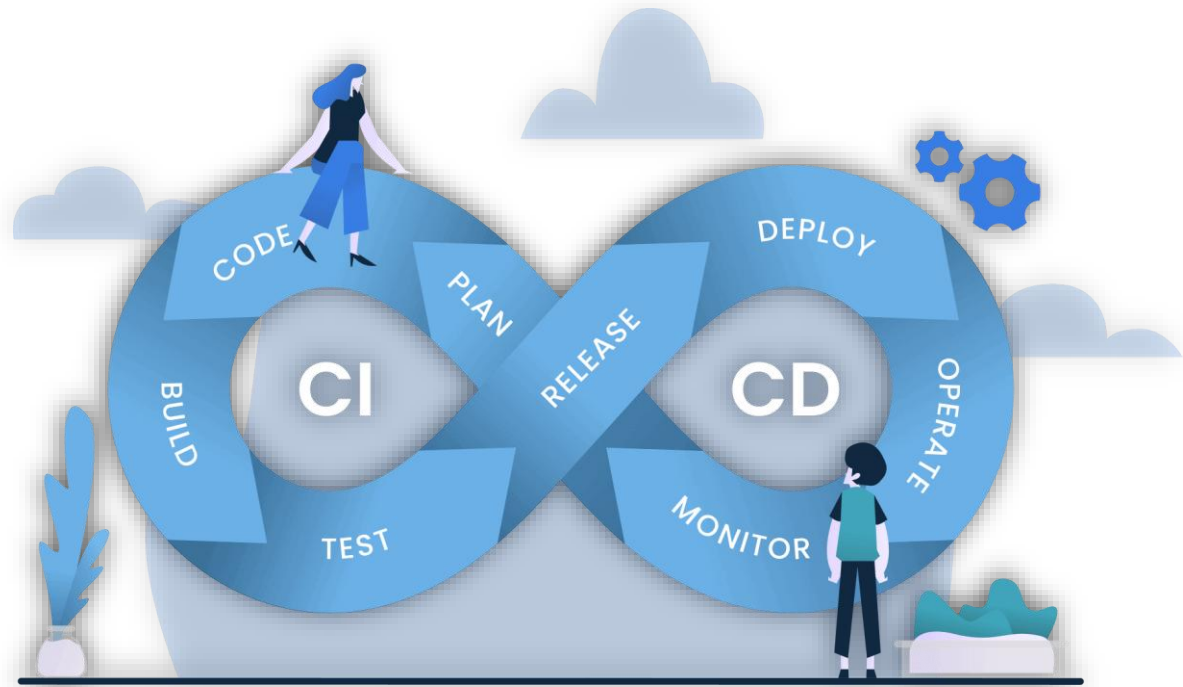


1st Testing

<Team 3>

박상우 위기화 김홍규

함신우 최희재



Contents index

- **Specification Review**
- **Category Partition Test**
- **Brute Force Test**

Specification Review -1003. Define Requirements

1003. Define Requirements

- **Functional Requirements**

- 사용자가 **결제** 요청으로 입력하는 카드를 검증해야 한다.
 - 카드번호가 존재하지 않거나, 카드 잔액이 부족한 카드인 경우 결제를 막아야 한다.
 - 그 외의 경우는 유효한 카드이므로 결제를 허용한다.
- 사용자가 해당 자판기에 재고가 있는 음료를 선택 후 결제하면, 해당 음료의 재고와 유저의 잔액을 차감해야한다.



1003. Define Requirements

- **Functional Requirements**

- 사용자가 **결제** 요청으로 입력하는 카드를 검증해야 한다.
 - 카드번호가 존재하지 않거나, 카드 잔액이 부족한 카드인 경우 결제를 막아야 한다.
 - 그 외의 경우는 유효한 카드이므로 결제를 허용한다.
- 사용자가 해당 자판기에 재고가 있는 음료를 선택 후 결제에 성공하면, 해당 음료의 재고와 유저의 잔액을 차감해야한다.

음료 재고, 카드 잔액의 차감조건 구체적으로 명시

Specification Review -1003. Define Requirements

1003. Define Requirements

● Functional Requirements

- 현재 자판기에는 재고가 없지만, 다른 자판기에는 재고가 있는 것이 확인된 음료에 한하여 현재 자판기에서 선결제를 할 수 있어야 한다.
 - 선결제를 요청하는 경우, 현재 자판기에서 결제 후 인증코드를 발급해야 한다.
 - 인증코드는 알파벳 대/소문자와 숫자를 포함하는 랜덤한 5자리 문자열이어야 한다.
 - 사용자가 선결제를 요청할 때마다 새로운 인증코드를 생성해야 한다.
 - 선결제가 완료된 음료의 경우, 음료를 지급할 자판기에서는 해당 음료를 이미 판매된 것으로 처리 후 재고를 차감해야 한다.
 - 선결제를 마친 사용자에게 인증코드와 선결제를 마친 자판기의 x좌표, y좌표로 구성된 위치 정보를 제공해야 한다.



1003. Define Requirements

● Functional Requirements

- 현재 자판기에는 재고가 없지만, 다른 자판기에는 재고가 있는 것이 확인된 음료에 한하여 현재 자판기에서 선결제를 할 수 있어야 한다.
 - 선결제의사를 확인해야 한다.
 - 선결제 의사가 없다면 초기화면 (메뉴화면)으로 돌아간다.
 - 선결제 의사가 있다면 아래를 진행한다.
 - 선결제를 요청하는 경우, 현재 자판기에서 결제 후 인증코드를 발급해야 한다.
 - 인증코드는 알파벳 대/소문자와 숫자를 포함하는 랜덤한 5자리 문자열이어야 한다.
 - 사용자가 선결제를 요청할 때마다 새로운 인증코드를 생성해야 한다.
 - 선결제가 완료된 음료의 경우, 음료를 지급할 자판기에서는 해당 음료를 이미 판매된 것으로 처리 후 재고를 차감해야 한다.
 - 선결제를 마친 사용자에게 인증코드와 선결제를 마친 자판기의 x좌표, y좌표로 구성된 위치 정보를 제공해야 한다.

● 선결제 의사가 없을 때 어떻게 진행할지에 대한 설명 추가

Specification Review - 1007. Define Business Use Case

1007. Define Business Use Case

Use Cases By Actor

- User
 - 결제 요청 (Payment Request)
 - 음료 선택 (Select Beverage)
 - 선결제 진행 (Pre-Payment)
 - 선결제 코드 입력 후 음료 수령 (Redeem Pre-Payment Code)
 -
- User Credit Repository
 - 카드 잔액 확인 (Validate Balance)
 - 유효한 카드 번호 확인 (Validate Credit Card)
- Another Vending Machine System
 - 재고 확인 응답 (Response Stock)
 - 선결제 응답 (Response Pre-Payment)
- None
 - 재고 확인 (Validate Stock)
 - 재고 차감 (Deduct Stock)
 - 결제 승인 (Approve Payment)
 - 결제 실패 (Reject Payment)
 - 재고가 있고 가까운 자판기 탐색 (Find Nearest Vending Machine)



1007. Define Business Use Case

Use Cases By Actor

- User
 - 결제 요청 (Payment Request)
 - 음료 선택 (Select Beverage)
 - 선결제 진행 (Pre-Payment)
 - 일반결제 진행 (Payment)
 - 결제 코드 입력 후 음료 수령 (Redeem Pre-Payment Code)
- User Credit Repository
 - 카드 잔액 확인 (Validate Balance)
 - 유효한 카드 번호 확인 (Validate Credit Card)
- Another DVM
 - 재고 확인 응답 (Response Stock)
 - 선결제 응답 (Response Pre-Payment)
- DVM
 - 재고 확인 (Validate Stock)
 - 재고 차감 (Deduct Stock)
 - 결제 승인 (Approve Payment)
 - 결제 실패 (Reject Payment)
 - 재고가 있고 가까운 자판기 탐색 (Find Nearest Vending Machine)

● 일반 결제에 대한 Case 설명 추가

● Use case들의 Actor 구체적으로 명시

Specification Review -1007. Define Business Use Case

1007. Define Business Use Case

- Describe use cases

Use Case	재고 확인 (Validate Stock)
Actor	<u>None</u>
Description	- 음료를 재고를 확인한다.

Use Case	재고 차감 (Deduct Stock)
Actor	<u>None</u>
Description	- 음료를 재고를 차감한다.

1007. Define Business Use Case

- Describe use cases

Use Case	결제 승인 (Approve Payment)
Actor	<u>None</u>
Description	- 사용자가 카드 번호를 입력한다. - 잔고 확인 후 결제를 승인한다.

Use Case	결제 실패 (Reject Payment)
Actor	<u>None</u>
Description	- 사용자가 카드 번호를 입력한다. - 유효한 카드가 아니라면 결제가 실패한다.

1007. Define Business Use Case

- Describe use cases

Use Case #05	재고 확인 (Validate Stock)
FR #	FR #1.1
Actor	<u>DVM</u>
Description	- 음료를 재고를 확인한다.

Use Case #06	재고 차감 (Deduct Stock)
FR #	FR # 1.2
Actor	<u>DVM</u>
Description	- 음료를 재고를 차감한다.

1007. Define Business Use Case

- Describe use cases

Use Case #07	결제 승인 (Approve Payment)
FR #	FR #2.1, FR #2.2, FR #2.3
Actor	<u>DVM</u>
Description	- 사용자가 카드 번호를 입력한다. - 잔고 확인 후 결제를 승인한다.

Use Case #08	결제 실패 (Reject Payment)
FR #	FR #2.1, FR #2.2, FR #2.3
Actor	<u>DVM</u>
Description	- 사용자가 카드 번호를 입력한다. - 유효한 카드가 아니라면 결제가 실패한다.



앞페이지의 내용 반영

Specification Review – 2031. Define Essential Use Case

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 <u>음료의 재고가 사용자로부터 받은 수량보다 크거나 같은지</u> 확인한다. 4. (D) 음료의 <u>재고가 있다면</u> 사용자에게 결제 요청 화면을 보여준다.
Alternative Courses of Events	1. (D) 음료의 재고가 <u>없다면</u> 다른 모든 DVM에게 해당 음료의 재고를 요청하여 확인한다. 2. (D) DVM은 <u>자신과 가장 가깝고</u> , 사용자가 구매하고자 하는 음료의 <u>재고가 있는</u> 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.

● ‘재고가 있다 or 없다’ -> ‘재고가 수량보다 ~하다’ 구체적으로 변경 및 통일
(변경 내용 다음 페이지에)

Specification Review - 2031. Define Essential Use Case

2031. Define Essential 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 사용자가 요청한 것 보다 크거나 같다면, 사용자에게 결제 요청 화면을 보여준다.
Alternative Courses of Events	1. (D) 음료의 재고가 사용자가 요청한것보다 작다면, 다른 모든 DVM에게 해당 음료의 재고를 요청하여 확인한다. 2. (D) DVM은 자신과 가장 가깝고, 음료의 재고가 사용자가 요청한 것보다 크거나 같은 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.

Specification Review - 2031. Define Essential Use Case

2141. Design Real Use Case

Use Case	UC-04. 인증 코드 입력 후 음료 수령
Actor	User
Purpose	사용자는 인증코드를 입력한 후 선결제 한 음료를 수령한다.
Overview	사용자가 인증코드를 입력하면, 유효성 확인 후 음료를 제공한다.
Type	Primary
Cross Reference	FR-06
Pre-Requisites	<u>UC-03 완료</u> , 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.

● 선결제 코드는 다른 자판기에서 받는 것이므로 조건에 명시하도록 변경
(변경 내용 다음 페이지에)

Specification Review - 2031. Define Essential Use Case

2031. Define Essential 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	<u>Other DVM에서 UC-03 완료</u> , 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.

Specification Review - 2031. Define Essential Use Case

2141. Design Real Use Case

Use Case	UC-05. 선결제 시 재고 확인 응답
Actor	<u>Other DVM</u>
Purpose	DVM은 <u>다른 DVM</u> 의 재고 확인 요청에 대해 응답한다.
Overview	DVM은 재고 확인 요청에 해당하는 음료의 재고와 자신의 위치를 요청을 보낸 <u>다른 DVM</u> 에 응답한다.
Type	Primary
Cross Reference	FR-07
Pre-Requisites	사용자가 <u>다른 DVM</u> 에서 선결제 진행 중이다
Typical Course of Events	(O) : <u>Other DVM</u> (D) : DVM 1. (O) <u>다른 DVM</u> 은 DVM에게 특정 음료에 대한 재고 확인 요청을 전송한다. 2. (D) DVM은 <u>다른 DVM</u> 으로부터 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



명칭 통일하도록 변경
(변경 내용 다음 페이지에)

Specification Review - 2031. Define Essential Use Case

2031. Define Essential Use Case

Use Case	UC-05. 선결제 시 재고 확인 응답
Actor	<u>Other DVM</u>
Purpose	DVM은 <u>Other DVM</u> 의 재고 확인 요청에 대해 응답한다.
Overview	DVM은 재고 확인 요청에 해당하는 음료의 재고과 자신의 위치를 요청을 보낸 <u>Other DVM</u> 에 응답한다.
Type	Primary
Cross Reference	FR-07
Pre-Requisites	사용자가 <u>Other DVM</u> 에서 선결제 진행 중이다
Typical Course of Events	(O) : <u>Other DVM</u> (D) : DVM 1. (O) <u>Other DVM</u> 은 DVM에게 특정 음료에 대한 재고 확인 요청을 전송한다. 2. (D) DVM은 <u>Other DVM</u> 으로부터 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

CATEGORY-PARTITION TEST

Parameters:

Authcode Exist: //하드코딩된 test Authcode 존재

True. [property Authcode]

False.

Start Input:

Select Beverage. [property running]

Insert Prepayment Code. [if Authcode][property OfferBeverage]

Other inputs. [error]

Exit.

Drink Code:

1 ~ 20. [if running][property drinkselected]

0 < || > 20. [if running][error]

character. [if running][error]

Quantity:

<1. [if drinkselected][error]

1 ~ stock. [if drinkselected][property quantityselected]

>stock. [if drinkselected][error]

character. [if drinkselected][error]

Card Number:

Valid number. [if quantityselected][property validcard]

Invalid number. [error]

Balance Check:

Price <= balance. [if validcard][property OfferBeverage]

Price > balance. [error]

전체 test case : 384
Error constraint : 13
Property constraint : 5

Test Case 9 (Key = 1.1.1.2.1.1.)

Authcode Exist : True

Start Input : Select Beverage

Drink Code : 1 ~ 20

Quantity : 1 ~ stock

Card Number : Valid number

Balance Check : Price <= balance

Test Case 10 (Key = 1.2.0.0.0.0.)

Authcode Exist : True

Start Input : Insert Prepayment Code

Drink Code : <n/a>

Quantity : <n/a>

Card Number : <n/a>

Balance Check : <n/a>

Test Case 11 (Key = 1.4.0.0.0.0.)

Authcode Exist : True

Start Input : Exit

Drink Code : <n/a>

Quantity : <n/a>

Card Number : <n/a>

Balance Check : <n/a>

Test Case 12 (Key = 2.1.1.2.1.1.)

Authcode Exist : False

Start Input : Select Beverage

Drink Code : 1 ~ 20

Quantity : 1 ~ stock

Card Number : Valid number

Balance Check : Price <= balance

Test Case 13 (Key = 2.4.0.0.0.0.)

Authcode Exist : False

Start Input : Exit

Drink Code : <n/a>

Quantity : <n/a>

Card Number : <n/a>

Balance Check : <n/a>

CATEGORY-PARTITION TEST

Test Case 9 (Key = 1.1.1.2.1.1.)

Authcode Exist : True
Start Input : Select Beverage
Drink Code : 1 ~ 20
Quantity : 1 ~ stock
Card Number : Valid number
Balance Check : Price <= balance

Test Case 10 (Key = 1.2.0.0.0.0.)

Authcode Exist : True
Start Input : Insert Prepayment Code
Drink Code : <n/a>
Quantity : <n/a>
Card Number : <n/a>
Balance Check : <n/a>

Test Case 11 (Key = 1.4.0.0.0.0.)

Authcode Exist : True
Start Input : Exit
Drink Code : <n/a>
Quantity : <n/a>
Card Number : <n/a>
Balance Check : <n/a>

Test Case 12 (Key = 2.1.1.2.1.1.)

Authcode Exist : False
Start Input : Select Beverage
Drink Code : 1 ~ 20
Quantity : 1 ~ stock
Card Number : Valid number
Balance Check : Price <= balance

Test Case 13 (Key = 2.4.0.0.0.0.)

Authcode Exist : False
Start Input : Exit
Drink Code : <n/a>
Quantity : <n/a>
Card Number : <n/a>
Balance Check : <n/a>

	Autocode	Start Input	Bev code	Quantity	Card Number	Card Balance	P/F
Case 9	True	Beverage	1~20	1~stock	Valid	Price <= Bal	P
Case 10	True	Prepayment code	-	-	-	-	P
Case 11	True	exit	-	-	-	-	P
Case 12	False	Beverage	1~20	1~stock	Valid	Price <= Bal	P
Case 13	False	exit	-	-	-	-	P

BRUTE FORCE TEST

Test	Test #	Description	Expected output	Test output	P / F
상품 목록, 메뉴 출력	1 - 1	ID, 상품명, 가격, 재고, 메뉴 정상 출력 여부 확인	ID, 상품명, 가격, 재고, 메뉴	ID, 상품명, 가격, 메뉴	F
	1 - 2	1을 눌러 음료 선택	음료 선택 메뉴로 이동	음료 선택 메뉴로 이동	P
	1 - 3	0을 눌러 종료	프로그램 종료	프로그램 종료	P
	1 - 4	<0 && >2의 정수 입력	에러 메시지 출력, 메뉴 선택으로 복귀	에러 메시지 출력, 메뉴 선택으로 복귀	P
	1 - 5	실수 입력	에러 메시지 출력, 메뉴 선택으로 복귀	에러 메시지 무한 출력	F
	1 - 6	문자 입력	에러 메시지 출력, 메뉴 선택으로 복귀	프로그램 종료	F
	1 - 7	문자열 입력	에러 메시지 출력, 메뉴 선택으로 복귀	프로그램 종료	F
음료 선택	2 - 1	유효하지 않은 정수 ID 입력	에러 메시지 출력, 음료 선택으로 복귀	수량 입력으로 진행	F
	2 - 2	유효하지 않은 실수 ID 입력	에러 메시지 출력, 음료 선택으로 복귀	에러 메시지 무한 출력	F
	2 - 3	유효하지 않은 문자 ID 입력	에러 메시지 출력, 음료 선택으로 복귀	에러 메시지 무한 출력	F
	2 - 3	유효하지 않은 문자열 ID 입력	에러 메시지 출력, 음료 선택으로 복귀	에러 메시지 무한 출력	F
	2 - 4	유효한 음료 ID 입력	수량 입력으로 진행	수량 입력으로 진행	P

BRUTE FORCE TEST

Test	Test #	Description	Expected output	Test output	P / F
수량 선택	3 - 1	0 입력	에러 메시지 출력, 수량 선택으로 복귀	에러 메시지 출력, 메뉴 선택으로 복귀	P
	3 - 2	음수 입력	에러 메시지 출력, 수량 선택으로 복귀	에러 메시지 출력, 메뉴 선택으로 복귀	P
	3 - 3	재고 초과 정수 입력	가장 가까운 다른 자판기 위치 안내	가장 가까운 다른 자판기 위치 안내	P
	3 - 4	1 이상, 재고 이내의 정수 입력	카드 번호 입력 메뉴 이동	카드 번호 입력 메뉴 이동	P
	3 - 5	1 이상, 재고 이내의 실수 입력	에러 메시지 출력, 수량 선택으로 복귀	소수점 자리가 카드 번호로 입력	F
	3 - 6	문자 입력	에러 메시지 출력, 수량 선택으로 복귀	에러 메시지 무한 출력	F
카드 번호 입력	4 - 1	잔고가 충분한 상태의 유효한 카드 번호 입력	카드 결제 성공, 카드 잔고 차감	카드 결제 성공, 카드 잔고 차감	P
	4 - 2	잔고가 부족한 상태의 유효한 카드 번호 입력	카드 결제 실패, 메뉴 선택으로 복귀	카드 결제 실패, 메뉴 선택으로 복귀	P
	4 - 3	유효하지 않은 카드 번호 입력	3회 실패 시 에러 메시지 출력 후 메뉴 선택으로 복귀	3회 실패 시 에러 메시지 출력 후 메뉴 선택으로 복귀	P
	4 - 4	유효하지 않은 카드 번호 1~2회 입력 후 잔고가 충분한 상태의 유효한 카드 번호 입력	카드 결제 성공, 카드 잔고 차감	카드 결제 성공, 카드 잔고 차감	P
	4 - 5	유효하지 않은 카드 번호 1~2회 입력 후 잔고가 충분하지 않은 유효한 카드 번호 입력	카드 결제 실패, 메뉴 선택으로 복귀	카드 결제 실패, 메뉴 선택으로 복귀	P

#Test Result = 10/23 = 43.4%

ISSUE TRACKING - JIRA

프로젝트 /  ku-team3 /  상위 항목 추가 /  TEAM3-12

1차 V&V

+ 추가

설명

Github Issue Link

- <https://github.com/DVM-System-2025-T03/CICD-Team3/issues/30>

상위 작업 (Ticket Number)

Step	Description	Description	Expected output	Test output	P / F
시작화면	1 - 1	시작 화면에서 's' 입력	메뉴 출력	메뉴 출력	
	1 - 2	1을 눌러 음료 선택	음료 선택 메뉴로 이동	음료 선택 메뉴로 이동	
	1 - 3	0을 눌러 종료	프로그램 종료	프로그램 종료	
	1 - 4	<0 && >2의 정수 입력	에러 메시지 출력, 메뉴 선택으로 복귀	에러 메시지 출력, 메뉴 선택으로 복귀	
	1 - 5	실수 입력	에러 메시지 출력, 메뉴 선택으로 복귀	에러 메시지 무한 출력	
	1 - 6	문자 입력	에러 메시지 출력, 메뉴 선택으로 복귀	프로그램 종료	
	1 - 7	문자열 입력	에러 메시지 출력, 메뉴 선택으로 복귀	프로그램 종료	
음료 선택	2 - 1	유효하지 않은 정수 ID 입력	에러 메시지 출력, 음료 선택으로 복귀	수량 입력으로 진행	
	2 - 2	유효하지 않은 실수 ID 입력	에러 메시지 출력, 음료 선택으로 복귀	에러 메시지 무한 출력	
	2 - 3	유효하지 않은 문자 ID 입력	에러 메시지 출력, 음료 선택으로 복귀	에러 메시지 무한 출력	