

2nd Testing

<system test & static analysis>

구상준, 류다은, 노현용, 김경희

Contents

Category Partition Test

Brute Force Test

Sonar Cloud

1. Category Partition Test

Environments: 시스템 동작 흐름 관련 조건들

항목명	설명
Initial Menu Selection	사용자가 자판기에서 처음 선택하는 메뉴- <code>Purchase</code> : 일반 구매- <code>Prepayment</code> : 사전 결제 음료 수령 (<code>property Prepayment</code>)
Stock Status	현재 자판기 재고 상태- <code>Sufficient</code> : 재고 충분 (<code>property Sufficient</code>)- <code>Insufficient</code> : 재고 부족, 방송 필요 (<code>BroadcastRequired</code> , <code>Insufficient</code>)
Broadcast to Other DVMs	재고가 부족할 경우, 다른 자판기(DVM)로 방송 여부- 조건: <code>[if BroadcastRequired]</code>
Other DVM Response	다른 자판기의 응답 유무- <code>Available</code> : 재고 응답 도착- <code>None_Responded</code> : 응답 없음 (에러)
Nearest DVM Selection	여러 DVM이 응답 시 선택 방식- <code>One_Candidate</code> : 하나 선택- <code>Tie_Lowest_ID</code> : 동일 거리 → ID 낮은 쪽 선택
Prepay Option Availability	재고 부족 시 선불 옵션을 제공할 수 있는지- <code>Enabled</code> : <code>Insufficient</code> 일 때만 허용됨- <code>Disabled</code> : 재고 충분 시
Payment Gateway Status	결제 서버의 온라인 상태- <code>Offline</code> : 에러
JSON Format Validity	통신용 JSON 포맷 유효성- <code>Malformed</code> : 에러 발생
Socket Communication Status	서버와의 소켓 연결 상태- <code>Timeout</code> : 연결 실패 시 에러
Payment Completion Time	결제 완료까지 소요된 시간- <code>>3s</code> : 시간이 오래 걸리면 에러 처리
Broadcast Response Time	다른 DVM으로부터의 응답 시간- <code>>1s</code> : 지연 시 에러
Concurrent Message Handling	자판기의 동시 메시지 처리 능력- <code>Overload</code> : 과부하 시 에러 발생

Parameters: 사용자 입력 및 처리 데이터

항목명	설명
Drink Code	음료 코드의 유효성- <code>invalid code</code> : 존재하지 않거나 비정상
Quantity Input	수량 입력값의 형식- <code>invalid format</code> : 숫자가 아니거나 허용 범위 초과
Card Format	카드 번호 형식 확인- <code>invalid format</code> : 숫자 길이 오류 등
Card Usability	카드 결제 가능 여부- <code>usable card</code> : 결제 가능- <code>unusable card</code> : 오류 처리
Payment Result	결제 서버의 처리 결과- <code>success</code> : 정상 결제- <code>insufficient_balance</code> , <code>server_error</code> : 에러 처리
Certification Code Format	사전 결제 인증 코드의 형식- <code>invalid</code> : 포맷 오류, <code>[if Prepayment]</code> 조건에만 발생
Certification Code Usage	인증 코드의 사용 여부- <code>already_used</code> : 이미 사용된 코드
Certification Code Registration	시스템에 등록된 코드인지 여부- <code>not_registered</code> : 미등록 에러
Prepaid Drink Availability	해당 코드로 수령 가능한 음료가 존재하는지 여부- <code>unavailable</code> : 에러 처리

테스트 코드

Environments:

Initial Menu Selection:

Purchase.	[property Purchase]
Prepayment.	[property Prepayment]
Invalid Format.	[error]
Multiple Tokens.	[error]
Out of Range.	[error]

Stock Status:

Sufficient.	[property Sufficient]
Insufficient.	[property BroadcastRequired] [property Insuffic

Broadcast to Other DVMs:

Yes	[if BroadcastRequired]
No	[if !BroadcastRequired]

Other DVM Response:

Available.	[if BroadcastRequired]
None_Responded.	[error] [if BroadcastRequired]

Nearest DVM Selection:

One_Candidate.	[if BroadcastRequired]
Tie_Lowest_ID.	[if BroadcastRequired]

Prepay Option Availability:

Enabled.	[if Insufficient]
Disabled.	[if Sufficient]

Payment Gateway Status:

Online.	
Offline.	[error]

JSON Format Validity:

Valid.	
Malformed.	[error]

```

Socket Communication Status:
Connected.
Timeout.                                [error]

Payment Completion Time:
<=3s
>3s                                    [error]

Broadcast Response Time:
<=1s
>1s                                    [error]

Concurrent Message Handling:
Success.
Overload.                              [error]

Parameters:
Drink Code:
valid code.
invalid code.                          [error]

Quantity Input:
valid quantity.
invalid format.                        [error]

Card Format:
valid format.
invalid format.                        [error]

Card Usability:
usable card.                           [property usable_card]
unusable card.                         [error]

Payment Result:
success.                               [if usable_card]
insufficient_balance.                  [error] [if usable_card]
server_error.                         [error] [if usable_card]

```

Certification Code Format:

valid.	[if Prepayment]
invalid.	[error] [if Prepayment]

Certification Code Usage:

unused.	[if Prepayment]
already_used.	[error] [if Prepayment]

Certification Code Registration:

registered.	[if Prepayment]
not_registered.	[error] [if Prepayment]

Prepaid Drink Availability:

available.	[if Prepayment]
unavailable.	[error] [if Prepayment]

테스트 케이스 작성

```
Test Case 1      <error>
  Other DVM Response :  None_Responded

Test Case 2      <error>
  Payment Gateway Status :  Offline

Test Case 3      <error>
  JSON Format Validity :  Malformed

Test Case 4      <error>
  Socket Communication Status :  Timeout

Test Case 5      <error>
  Concurrent Message Handling :  Overload

Test Case 6      <error>
  Drink Code :  invalid code

Test Case 7      <error>
  Quantity Input :  invalid format

Test Case 8      <error>
  Card Format :  invalid format

Test Case 9      <error>
  Card Usability :  unusable card
```

Test Case 10 <error>

Payment Result : insufficient_balance

Test Case 11 <error>

Payment Result : server_error

Test Case 12 <error>

Certification Code Format : invalid

Test Case 13 <error>

Certification Code Usage : already_used

Test Case 14 <error>

Certification Code Registration : not_registered

Test Case 15 <error>

Prepaid Drink Availability : unavailable

Test Case 16 (Key = 1.1.0.0.2.1.1.1.1.1.1.1.1.0.0.0.0.)

Initial Menu Selection	:	Purchase
Stock Status	:	Sufficient
Other DVM Response	:	<n/a>
Nearest DVM Selection	:	<n/a>
Prepay Option Availability	:	Disabled
Payment Gateway Status	:	Online
JSON Format Validity	:	Valid
Socket Communication Status	:	Connected
Concurrent Message Handling	:	Success
Drink Code	:	valid code
Quantity Input	:	valid quantity
Card Format	:	valid format
Card Usability	:	usable card
Payment Result	:	success
Certification Code Format	:	<n/a>
Certification Code Usage	:	<n/a>
Certification Code Registration	:	<n/a>
Prepaid Drink Availability	:	<n/a>

Test Case 17 (Key = 1.2.0.0.1.1.1.1.1.1.1.1.1.0.0.0.0.)

Initial Menu Selection	:	Purchase
Stock Status	:	Insufficient
Other DVM Response	:	<n/a>
Nearest DVM Selection	:	<n/a>
Prepay Option Availability	:	Enabled
Payment Gateway Status	:	Online
JSON Format Validity	:	Valid
Socket Communication Status	:	Connected
Concurrent Message Handling	:	Success
Drink Code	:	valid code
Quantity Input	:	valid quantity
Card Format	:	valid format
Card Usability	:	usable card
Payment Result	:	success
Certification Code Format	:	<n/a>
Certification Code Usage	:	<n/a>
Certification Code Registration	:	<n/a>
Prepaid Drink Availability	:	<n/a>

Test Case 18 (Key = 1.2.1.1.1.1.1.1.1.1.1.1.1.1.0.0.0.0.)

```
Initial Menu Selection      : Purchase
Stock Status               : Insufficient
Other DVM Response        : Available
Nearest DVM Selection     : One_Candidate
Prepay Option Availability : Enabled
Payment Gateway Status     : Online
JSON Format Validity       : Valid
Socket Communication Status : Connected
Concurrent Message Handling : Success
Drink Code                 : valid code
Quantity Input             : valid quantity
Card Format                 : valid format
Card Usability             : usable card
Payment Result             : success
Certification Code Format   : <n/a>
Certification Code Usage   : <n/a>
Certification Code Registration : <n/a>
Prepaid Drink Availability : <n/a>
```

Test Case 19 (Key = 1.2.1.2.1.1.1.1.1.1.1.1.1.1.0.0.0.0.)

```
Initial Menu Selection      : Purchase
Stock Status               : Insufficient
Other DVM Response        : Available
Nearest DVM Selection     : Tie_Lowest_ID
Prepay Option Availability : Enabled
Payment Gateway Status     : Online
JSON Format Validity       : Valid
Socket Communication Status : Connected
Concurrent Message Handling : Success
Drink Code                 : valid code
Quantity Input             : valid quantity
Card Format                 : valid format
Card Usability             : usable card
Payment Result             : success
Certification Code Format   : <n/a>
Certification Code Usage   : <n/a>
Certification Code Registration : <n/a>
Prepaid Drink Availability : <n/a>
```

Test Case 20 (Key = 2.1.0.0.2.1.1.1.1.1.1.1.1.1.1.1.1.)

Initial Menu Selection	: Prepayment
Stock Status	: Sufficient
Other DVM Response	: <n/a>
Nearest DVM Selection	: <n/a>
Prepay Option Availability	: Disabled
Payment Gateway Status	: Online
JSON Format Validity	: Valid
Socket Communication Status	: Connected
Concurrent Message Handling	: Success
Drink Code	: valid code
Quantity Input	: valid quantity
Card Format	: valid format
Card Usability	: usable card
Payment Result	: success
Certification Code Format	: valid
Certification Code Usage	: unused
Certification Code Registration	: registered
Prepaid Drink Availability	: available

Test Case 21 (Key = 2.2.1.1.1.1.1.1.1.1.1.1.1.1.1.1.1.)

Initial Menu Selection	: Prepayment
Stock Status	: Insufficient
Other DVM Response	: Available
Nearest DVM Selection	: One_Candidate
Prepay Option Availability	: Enabled
Payment Gateway Status	: Online
JSON Format Validity	: Valid
Socket Communication Status	: Connected
Concurrent Message Handling	: Success
Drink Code	: valid code
Quantity Input	: valid quantity
Card Format	: valid format
Card Usability	: usable card
Payment Result	: success
Certification Code Format	: valid
Certification Code Usage	: unused
Certification Code Registration	: registered
Prepaid Drink Availability	: available

Test Case 22 (Key = 2.2.1.2.1.1.1.1.1.1.1.1.1.1.1.1.1.)

Initial Menu Selection	: Prepayment
Stock Status	: Insufficient
Other DVM Response	: Available
Nearest DVM Selection	: Tie_Lowest_ID
Prepay Option Availability	: Enabled
Payment Gateway Status	: Online
JSON Format Validity	: Valid
Socket Communication Status	: Connected
Concurrent Message Handling	: Success
Drink Code	: valid code
Quantity Input	: valid quantity
Card Format	: valid format
Card Usability	: usable card
Payment Result	: success
Certification Code Format	: valid
Certification Code Usage	: unused
Certification Code Registration	: registered
Prepaid Drink Availability	: available

테스트 흐름 작성

Test Case 16:

정상적인 기본 흐름

- 메뉴: 일반 구매 (**Purchase**)
 - 재고: 충분 (**Sufficient**)
 - 방송 없음, 인증 코드 없음
 - 결제 및 카드 정보 모두 정상
-

Test Case 17:

재고 부족알람 후 메인으로 이동

- 재고 상태가 **Insufficient**
 - 메인으로 이동
-

Test Case 18:

재고 부족으로 인해 다른 자판기 요청 발생

- 재고 상태가 **Insufficient** → 다른 자판기로 방송이 이루어짐
 - 다른 DVM의 응답을 기다리고, Prepay 옵션이 **활성화됨 (Enabled)**
 - 인증 코드 흐름은 없음
 - 카드 및 결제 모두 정상
-

Test Case 19:

여러 DVM 응답 중 가장 가까운 자판기 선택 (ID 기준 타이 브레이크)

- Test Case 17과 유사하지만
 - 여러 응답 중 **Tie_Lowest_ID** 전략으로 DVM 선택 수행
-

Test Case 20:

Prepayment 인증 코드 흐름 테스트 (정상 수형)

- 메뉴가 Prepayment로 변경됨
- 인증 코드 관련 검사 항목들이 활성화됨
 - 포맷 유효성 검사
 - 사용 여부 확인
 - 등록 여부 확인
 - 사전 결제된 음료의 제공 가능 여부 확인
- 재고는 충분하여 방송 없음
- 결제는 정상

Test Case 21:

Prepayment + 재고 부족 + 다른 DVM 요청 흐름 결합

- 메뉴: Prepayment
- 재고 부족 → 방송 요청 발생 및 DVM 응답 처리
- 인증 코드 관련 모든 항목이 정상적으로 처리됨
- Prepay 옵션도 활성화됨

Test Case 22:

Prepaid + 재고 부족 + Tie 상황의 DVM 선택 흐름 결합

- Test Case 20과 동일한 상황에서
- 가장 가까운 자판기 선택 시 Tie_Lowest_ID 전략을 사용함

테스트 과정

cmake 부분

```

/home/ubuntu/advancedsw/src/app/application/dvm.cpp: In member function 'std::string DVM::queryStocks(std::string, int)':
/home/ubuntu/advancedsw/src/app/application/dvm.cpp:52:32: error: 'INT_MAX' was not declared in this scope
   52 |     int shortestDistance = INT_MAX;
      |                                ^
/home/ubuntu/advancedsw/src/app/application/dvm.cpp:2:1: note: 'INT_MAX' is defined in header '<climits>'; did you forget to '#include <climits>'?
   1 | #include "dvm.h"
     | ^
++ | #include <climits>
   2 |
/home/ubuntu/advancedsw/src/app/application/dvm.cpp:1:1: error: 'INT_MAX' is defined in header '<climits>'; did you forget to '#include <climits>'?

```

advancedsw/src/app/application/dvm.cpp

#include <climits> 추가 요청

INT_MAX 오류 climits는 C++ 사용되는 표준헤더임

```

/home/ubuntu/advancedsw/src/app/presentation/controller.cpp:133:65: error: no matching function for call to 'max()'
   133 |     cin.ignore(std::numeric_limits<std::streamsize>::max(), '\n');
      |                                ^
In file included from /usr/include/c++/13/string:51,
                 from /usr/include/c++/13/bits/locale_classes.h:40,
                 from /usr/include/c++/13/bits/ios_base.h:41,
                 from /usr/include/c++/13/ios:44,
                 from /usr/include/c++/13/ostream:40,
                 from /usr/include/c++/13/istream:41,
                 from /home/ubuntu/advancedsw/src/app/presentation/external/cord.h:4,
                 from /home/ubuntu/advancedsw/src/app/presentation/controller.h:4,
                 from /home/ubuntu/advancedsw/src/app/presentation/controller.cpp:1:
/usr/include/c++/13/bits/stl_algobase.h:257:18: note: candidate: 'template<class Tp, constmp const Tp& std::max(const Tp&, const Tp&)'
   257 |     max(const Tp& __a, const Tp& __b)
      |     ~~~~
/home/ubuntu/advancedsw/src/app/presentation/controller.cpp:133:65: note: template argument deduction/substitution failed:
/home/ubuntu/advancedsw/src/app/presentation/controller.cpp:133:65: note: candidate expects 2 arguments, 0 provided
   133 |     cin.ignore(std::numeric_limits<std::streamsize>::max(), '\n');
      |                                ^
/usr/include/c++/13/bits/stl_algobase.h:303:5: note: candidate: 'template<class Tp, class Compare> constmp const Tp& std::max(const Tp&, const Tp&, const Compare&)'
   303 |     max(const Tp& __a, const Tp& __b, Compare __comp)
      |     ~~~~
/usr/include/c++/13/bits/stl_algobase.h:388:18: note: template argument deduction/substitution failed:
/home/ubuntu/advancedsw/src/app/presentation/controller.cpp:133:65: note: candidate expects 3 arguments, 0 provided
   133 |     cin.ignore(std::numeric_limits<std::streamsize>::max(), '\n');
      |                                ^
In file included from /usr/include/c++/13/range:52,
                 from /home/ubuntu/advancedsw/src/app/presentation/controller.h:10:
/usr/include/c++/13/bits/stl_algo.h:5795:18: note: candidate: 'template<class Tp, constmp const Tp& std::max(initializer_list<Tp>)'
   5795 |     max(initializer_list<Tp> __l)
      |     ~~~~
/usr/include/c++/13/bits/stl_algo.h:5795:5: note: template argument deduction/substitution failed:
/home/ubuntu/advancedsw/src/app/presentation/controller.cpp:133:65: note: candidate expects 1 argument, 0 provided
   133 |     cin.ignore(std::numeric_limits<std::streamsize>::max(), '\n');
      |                                ^
/usr/include/c++/13/bits/stl_algo.h:5805:5: note: candidate: 'template<class Tp, class Compare> constmp const Tp& std::max(initializer_list<Tp>, Compare)'
   5805 |     max(initializer_list<Tp> __l, Compare __comp)
      |     ~~~~
/home/ubuntu/advancedsw/src/app/presentation/controller.cpp:133:65: note: template argument deduction/substitution failed:
/home/ubuntu/advancedsw/src/app/presentation/controller.cpp:133:65: note: candidate expects 2 arguments, 0 provided
   133 |     cin.ignore(std::numeric_limits<std::streamsize>::max(), '\n');
      |                                ^

```

advancedsw/src/app/presentation/controller.cpp

#include <limits> 추가요청

기존 메인 메뉴에서 "01 10"을 치면

다음 음료수 화면에서 "01 10"을 작성해도 이전 "10"이라는 정보가 넘어가서 오류 발생

```

1. 음료 구매 하기
2. 선결제 한 음료 받아 가기

Enter menu : 01 10

구매 하고 싶은 메뉴와 수량을 입력해 주세요 .
ex) 02 10

-----
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)
-----

Enter menu : 01 10
Connection failed

요청하신 음료 (10)는 현재 재고가 없습니다.

메인 화면으로 돌아갑니다.
계속하려면 Enter를 누르세요 ...

```

TestCase16

정상 케이스

▶ 1→01 10→1234-abcd

PASS

TaseCase17,18

재고 부족 케이스

▶ 1 → 01 30

DVM 부분도 똑같은 (실패)

올려준 PPT와 동일한 실험시 메뉴에서 "03 1" 작성시 최소 3분이상 타임 아웃 후 코드 실행됨

```

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

```

```

Enter menu : 03 1

Connection failed

요청하신 음료 (03)는 현재 재고가 없습니다.

메인 화면으로 돌아갑니다.
계속하려면 Enter를 누르세요 ...

안녕하세요, Team1 DVM입니다.

희망하는 옵션을 선택해주세요.

1. 음료 구매하기
2. 선결제 한 음료 받아가기

```

※추가 하면 좋은것: 다른 DVM의 응답 부분에서 타임아웃 필요

TestCase20

▶ 선결제 음료 오류(정상)

테스트 결과

각 sale_test, domain_test, integration_test, otherdvm_test, controller_test,dvm_test 전부 통과

TestCase16	PASS	정상
TestCase17	PASS	정상
TestCase18	PASS	정상
TestCase19	PASS	정상
TestCase20	PASS	정상
TestCase21	PASS	정상
TestCase22	PASS	정상

7/7

100% 달성

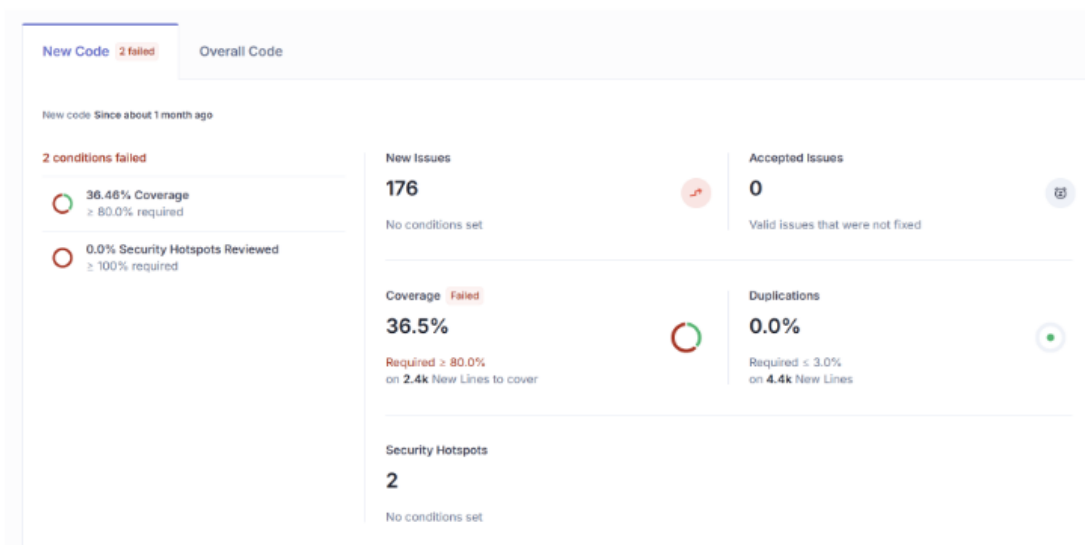
1. Brute Force Test

TestNum	Use Case	Test Description	Expected Output
BF-01	RUC 01 초기메뉴 선택	본술에 선택지를 보여주고, 사용자가 유효한 숫자(1 또는 2)를 입력할 때까지 반복 입력을 받는다.	1이나 2가 아니면 잘못 입력했다는 메시지와 함께 다시 입력하도록 해야함
BF-02	RUC 01 초기메뉴 선택	각 번호에 맞게 입력했을 때 올바르게 출력값이 나오는지 확인	1번 입력했을 경우 음료 구매 화면으로, 2번 입력 했을 경우 선결제 화면으로 이동
BF-03	RUC 02 음료 종류 및 수량 선택	유효하지 않은 번호 입력 시 결과 확인	음료 이외의 번호 입력 시 화면 변화없이 재입력 요청
BF-04	RUC 02 음료 종류 및 수량 선택	입력 한 후 초과시간 지났을 때 반응 확인	초기화면으로 돌아가야 함
BF-05	RUC 02 음료 종류 및 수량 선택	숫자가 아닌 문자열을 넣었을 때 반응 확인	문자열이 들어왔으니 화면 변화없이 재입력 요청 및 안정적 화면 출력
BF-06	RUC 02 음료 종류 및 수량 선택	수량을 0 및 음수로 입력했을 때 반응 확인	0과 음수 수량은 유효하지 않으므로 재입력 요청
BF-07	RUC 02 음료 종류 및 수량 선택	구매하고 싶은 수량은 자리가 있는지	수량이 너무 큰 값이 입력될 경우 최대값 이하로 입력해달라고 메시지 출력, 최대값이 몇으로 되어 있는가
BF-08	RUC 02 음료 종류 및 수량 선택	자판기에서 판매 가능한 현재 음료 아이템 목록을 출력한다.	1부터 20까지의 메뉴 display
BF-09	RUC 03 카드 정보 입력과 확인	음료 가격이 잘못게 계산되는지 확인	음료 가격 * 수량으로 문제 없음
BF-10	RUC 03 카드 정보 입력과 확인	유효한 카드정보와 유효하지 않은 카드정보 입력 시 결과 확인	유효한 카드 정보 입력시 결제가 완료되며, 유효하지 않은 카드 정보는 다시 입력 요청
BF-11	RUC 03 카드 정보 입력과 확인	카드번호에 공백, 특수문자가 포함 된 경우	잘못된 형식으로 간주되어 재입력 요청
BF-13	RUC 04 음료 판매 처리(현재 자판기)	결제 완료 후 Enter 누를 경우와 다른 key를 눌렀을 때 확인	Enter를 입력하면 메인 화면으로 돌아감
BF-14	RUC 04 음료 판매 처리(현재 자판기)	음료 결제 후 자판기에서 해당 수량만큼 차감되는지 확인	해당 수량만큼 차감 됨
BF-15	RUC 03 다른 자판기 조회 및 위치 안내	현재 자판기 및 다른 자판기의 재고 상태를 확인하고 본인의 재고가 충분할 경우 판매한다.	재고가 충분하지 않다면 가장 가까운 다른 판매기의 위치를 출력 및 선결제 진행
BF-16	RUC 05 다른 자판기 조회 및 위치 안내	재고 상태를 확인하고 충분하지 않을 경우 다른 자판기의 위치를 알려주거나 not available를 출력	다른 자판기의 위치나 not available 메시지 출력
BF-17	RUC 05 다른 자판기 조회 및 위치 안내	카드 정보 잘못 입력 시, 초기화면으로 돌아가는건 아닌지 확인	카드 정보 잘못 입력 시, 재입력 요청
BF-18	RUC 05 다른 자판기 조회 및 위치 안내	다른 자판기에도 재고가 없으면 어떤 메시지가 나오는지	재고가 없어서 판매할 수 없다는 메시지 출력
BF-19	RUC 05 다른 자판기 조회 및 위치 안내	재고가 있는 다른 자판기 위치가 제대로 출력되는지	(3, 4) 위치의 자판기에서 판매가능하다는 식의 메시지 출력
BF-20	RUC 05 다른 자판기 조회 및 위치 안내	인증코드가 정상적으로 5자리로 생성되는지 확인	영숫자 5자리 코드 생성
BF-21	RUC 05 다른 자판기 조회 및 위치 안내	구매 가능한 자판기 위치와 인증 코드 출력되는지	위치 (3, 4) 인증코드 736hr과 같은 형식으로 출력
BF-22	RUC 06 선결제 구입	특정 DVM 대상의 선결제를 처리하고 인증 코드를 반환한다.	선결제 금액을 Sale 객체에 판매 아이템 업데이트 동시에 총 판매 금액을 해당 금액 만큼 증가시켜준다.
BF-23	RUC 07 선결제 음료 수령	유효한 인증코드와 유효하지 않은 코드 입력 시 결과 확인	유효한 인증코드만 허가되어 음료 수령 가능
BF-24	RUC 07 선결제 음료 수령	음료 수령 시, 어떤 자판기에서 수량이 빠져나가는지 확인	선결제한 자판기에서 음료 수량이 차감 됨
BF-25	RUC 07 선결제 음료 수령	음료 제공 후 메인 화면으로 돌아갈 때 Enter가 아닌 다른 키가 입력했을 경우 확인	Enter를 입력하라는 메시지 출력
BF-26	RUC 07 선결제 음료 수령	인증코드를 입력받아 선결제된 음료를 수령 처리	sales 내에 일치하는 certCode가 있는 경우 수령 처리
BF-27	RUC 07 선결제 음료 수령	인증 코드를 한 번만 사용할 수 있도록 사용 여부 관리를 한다.	사용 여부를 boolean으로 잘 저장하고 있어야 함
BF-28	RUC 07 선결제 음료 수령	입력된 인증코드가 일치하고, 아직 사용되지 않았다면 사용 처리를 한다.	사용 처리 후 isUsed true로 업데이트

TestNum	pass/fail	비고
BF-01	pass	
BF-02	pass	음료 구매 화면에서 홈으로 돌아가는 방법이 있으면 좋을 듯
BF-03	pass	
BF-04	pass	
BF-05	pass	
BF-06	pass	
BF-07	pass	
BF-08	pass	수량, 가격 미출력
BF-09	pass	
BF-10	pass	
BF-11	pass	
BF-13	pass	
BF-14	pass	차감이 되는 것 같긴 하지만 육안으로는 확인 불가 (더이상 재고가 없다는 출력보고 임의 확인)
BF-15	pass	
BF-16	pass	
BF-17	pass	
BF-18	pass	
BF-19	pass	
BF-20	pass	
BF-21	pass	
BF-22	pass	
BF-23	pass	
BF-24	pass	
BF-25	pass	
BF-26	pass	
BF-27	pass	
BF-28	pass	

Pass rate = 100%

1. Sonar Cloud



실행 시 필요 라이브러리가 없어서 추가 후 실행 필요

Dvm.cpp <climit.h>

Controller.cpp <limit.h>

Coverage 36.5%

Software Quality		Severity		Clean Code Attribute	
Security	0	Blocker	1	Consistency	13
Reliability	0	High	45	Intentionality	134
Maintainability	176	Medium	102	Adaptability	29
		Low	27	Responsibility	0
		Info	1		

Type	Sub	Count
Bug	-	0
Vulnerability	-	0
Code Smell	Blocker	1
	High	45
	Medium	102
	Low	27
	Info	1

Type	Count
Bug	0
Vulnerability	0
Code Smell	176

File	Line	Severity	Review
dvm.cpp	60	High	코드에 if for do while switch 문을 3개 이상 중첩하지 않도록 주의필요
	12	Medium	Std::exception, std::logic_error와 같은 일반적인 예외처리를 할려고 하면 std::runtime_error와 같은 예외처리를 잡으려고 하는 모든 코드에 부정적인 영향이 미침
controller.cpp	13	Medium	map<string, string>부분에 const를 선언필요
	112	Medium	필요하지 않는 주석은 제거필요
	359	Medium	전역변수로 e를 사용했는데 사용되지 않는 예외 매개변수 제거필요
sale.cpp	15	High	time(0)가 아닌 nullptr를 사용필요
	16	Blocker	ctime 호출을 ctime_r로 변경 필요
prepayment.h	9	Medium	사용되지 않는 Private field 제거필요
	13	High	prepayment(변환 연산자)에 explicit 키워드가 필요함
otherdvm.cpp	69	Low	If문안에 init문을 사용해서 warnCount를 선언함
Location.h	10	Info	불필요한 주석 제거 필요

전체적으로 **const**를 사용할 때 비용이 드는 객체들을 사용하여 **Medium**쪽이 많이 구성됨