

2018 SOFTWARE VERIFICATION

# 2<sup>nd</sup> Testing

System Test, Static Analysis

2018.06.04 (월) SV T1

201311263 김민환

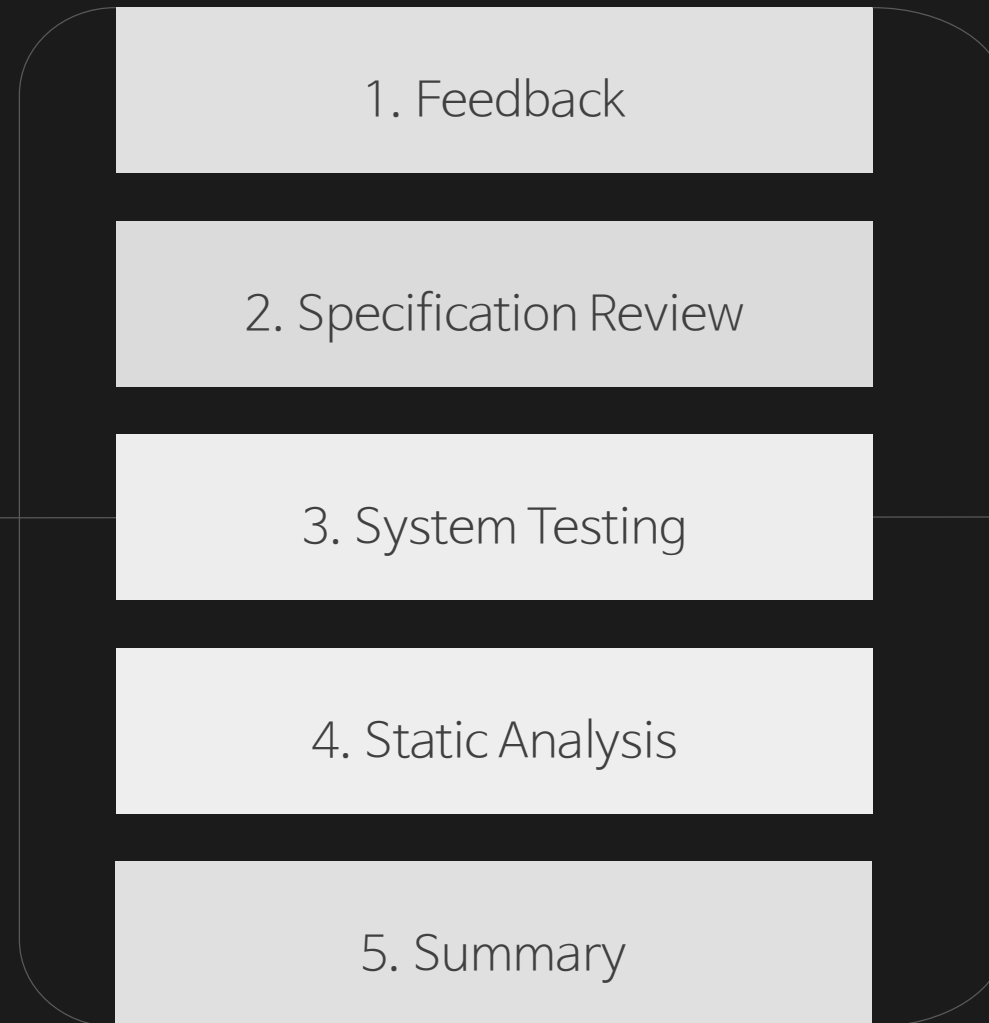
201311308 전세진

201411278 서희진

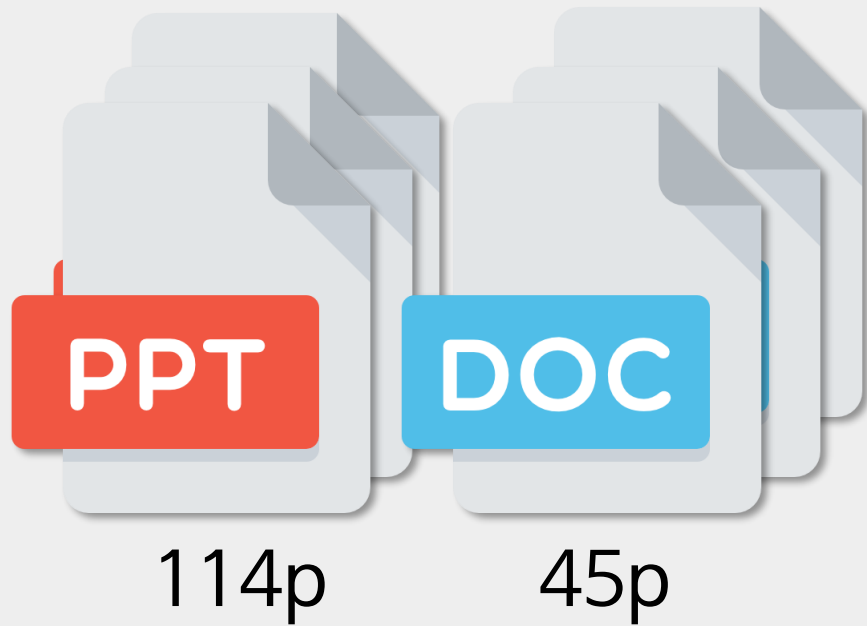
201411317 조민규

# 2<sup>nd</sup> Testing

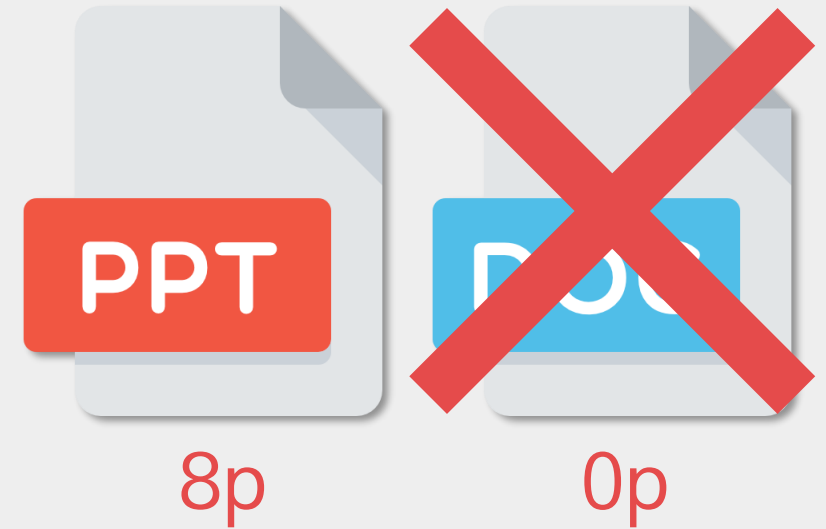
2018 SOFTWARE VERIFICATION



# 01. Feedback



소프트웨어 검증 팀



소프트웨어 모델링 팀

[2018 SMA T1] 5<sup>th</sup> 발표 2p

## 직관적이지 않은 인터페이스

8. Non Functional Requirement를 모두 만족시키지 못하였다.
  - A. 직관적이지 않은 UI
    - i. 한글 깨짐 현상 발생 (파일과 코드 전체가 EUC-KR로 저장되어있다.)
    - ii. Placeholder 처리가 되어있지 않다. (default text로 값이 들어가 있다.)
    - iii. Maxlength가 지정되어 있지 않다.
    - iv. Number format이 적용되어 있지 않다. (구분자가 없다.)
    - v. Return Card가 닫지 않으면 계속 출력된다.
    - vi. account id를 입력할 시 기존에 있던 label이 삭제되고 그 곳에 account id가 입력된다.

[2018 SV] 4<sup>th</sup> 보고서

1. 처음 받았을 당시 매뉴얼의 부재  
어떤 식으로 동작하는지 판별 불가

2. ItemType, ItemID, ItemBank..  
어떤 범위의 값을 갖는 지  
문서에 명시되어 있지 않음.

3. 카드를 넣을 때에도 계좌번호를 넣어주어야 함  
카드 '나' 통장을 넣어주세요..?

## 지난 발표 Brute Force Testing 결과

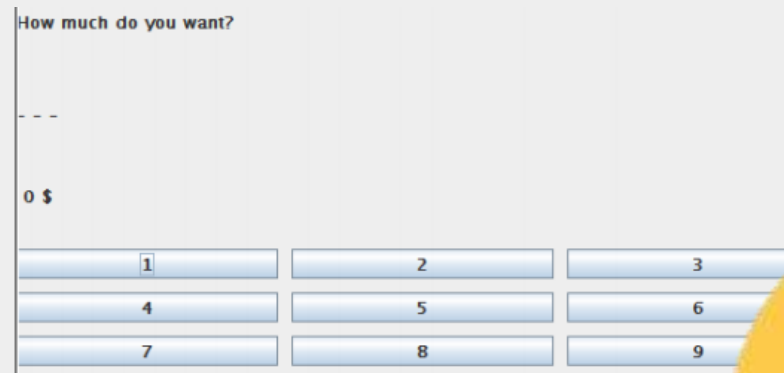


▲ 4-2. 외화를 출금 시 10배의 금액이 출금된다.

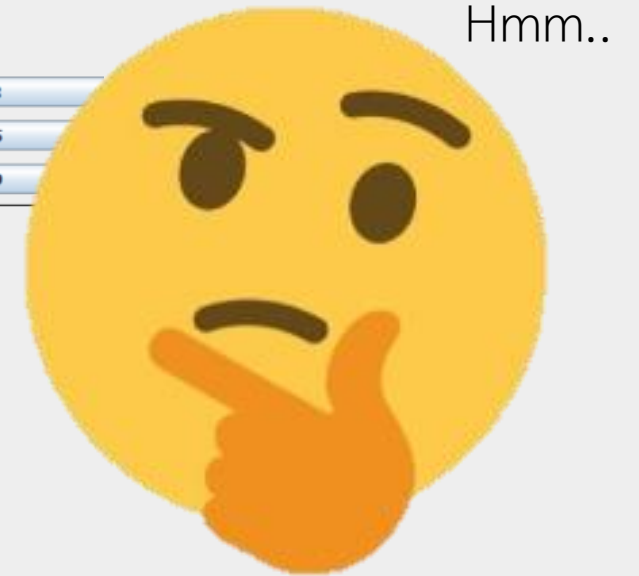
Transaction Amount : 3 \$

▲ 6-1.

교통카드의 발급비용은 0.3\$라고 명시되어 있지만 3\$가 출금된다.



▲ 4-3. 세 자리의 숫자만 출금이 가능하다.



# 예외처리 알람 환율정보

## 1. 예외처리

실행이 안되는 에러의 예외처리 필요

## 2. 알람

문제가 생겼을 때 알람이 오지 않는다  
= 문제가 생겨도 그냥 진행된다.

## 3. 환율 정보

지난 발표에서의 중요한 내용이 아님.

	팀발표 #5
	OOPT 2nd Cycle
	Static Analysis
	<u>발표자료</u> <u>소스코드</u> <u>메뉴얼</u>
	<u>발표자료</u> <u>다음보고서</u> <u>소스코드</u>

## 불친절한 발표자료

발표주제와 다른 내용만을 나열함

## + 대응 보고서의 부재

필수 사항은 아니지만

이전 단계 문서 수정 내역 파악 어려움



# 02. Specification Review

## 1003. Define Requirements

## 1. Functional Requirements

No.	Function	Description
1	Check	계좌 잔액 확인
2	Deposit	현금을 받아, 확인하고, 계좌에 증감
3	Withdraw	User 가 원하는 현금만큼 계좌에서 차감하여 현금으로 교환
4	Transfer	User 가 원하는 금액만큼 입력한 타계좌로 이동
5	Update	거래 내역 업데이트
6	Status Alarm	ATM 현금/교통카드/명세표 소지량 경고 알림
7	IssueTrafficcard	외국계좌와 연동된 교통카드 발급
8	Management	관리자가 ATM 안 현금을 관리한다.

→ 2. Deposit : 증감 → 증가

End Use Case 정의되어 있지 않음

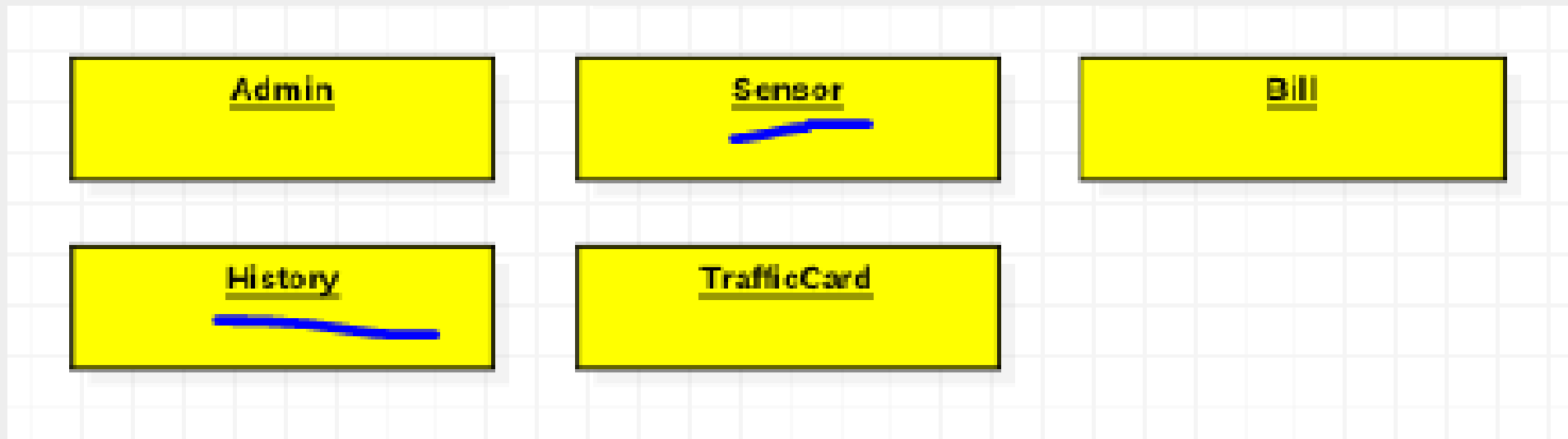
## 1004. Record Terms in Glossary

**TrafficCard**

ATM 에서 지급하는 TrafficCard(이용할 때마다 계좌에서 금액이 빠져 나간다.)

→ ATM에서 지급하는 구조가 아님, OOAD 개발방법론을 따랐는지 의문

## 1008. Define Business Concept Models



→ 불필요한 심볼의 정의(Sensor, History)

## 2031. Describe Use Case

## 1.1 Check

2030

Alternative Courses  
of EventsE0. 한국 계좌의 경우 한국어 모드, 해외 계좌일 경우 외국어 모드  
E7. 명세표 출력시, 계좌 잔액 출력 / 미출력시 다음 단계.Exceptional Courses  
of EventsE1. 인증에 실패한 경우, 종료한다  
E8. 명세표 출력 용지가 부족할 경우 알림을 발생시킨다.

2040

Alternative Courses  
of EventsE0. 한국 계좌의 경우 한국어 모드, 해외 계좌일 경우 외국어 모드  
E1. 다른 거래 선택  
E7. 명세표 출력시, 계좌 잔액 출력 / 미출력시 다음 단계.

→ 2030 과 2040, 2041 Use Case 불일치

## 2031. Describe Use Case

## 1.2 Deposit

2030

## Use Case

- E0. ATM 보유 현금이 초과되었을 경우 종료한다.
- E2. 사용자가 폐쇄요청을 하지 않을 경우, 일정 시간 후에 안내 후 자동 폐쇄한다.
- E9. 현금 보유량이 많을 경우 알림을 발생시킨다.
- E10. 명세표 출력 용지가 부족할 경우 알림을 발생시킨다.

2040

## Use Case

- E0. ATM 보유 현금이 초과되었을 경우 종료한다.
- E2. 사용자가 폐쇄요청을 하지 않을 경우, 일정 시간 후에 안내 후 자동 폐쇄한다.
- E4. 유효한 현금이 아닐 경우, 반환 후 종료한다.
- E9. 현금 보유량이 많을 경우 알림을 발생시킨다.
- E10. 명세표 출력 용지가 부족할 경우 알림을 발생시킨다.

→ 2030 과 2040, 2041 Use Case 불일치

## 2031. Describe Use Case

## 1.3 Withdraw

2030

## Use Case

E0. 한국 계좌의 경우 한국어 모드, 해외 계좌일 경우 외국어 모드  
E11. 명세표 출력 시, 거래 내역 출력 / 미출력시 다음 단계.

2040

## Use Case

E0. 한국 계좌의 경우 한국어 모드, 해외 계좌일 경우 외국어 모드  
E0. 다른 거래 선택  
E11. 명세표 출력 시, 거래 내역 출력 / 미출력시 다음 단계.

→ 2030 과 2040, 2041 Use Case 불일치

## 2031. Describe Use Case

## 1.3 Withdraw

2030

**Actors**

- E1. 본인확인 실패 시, 종료한다.
- E5. 계좌 잔액 부족 시, 종료한다.
- E11. 현금 인출이 일정 시간동안 이루어지지 않는 경우 투입구를 폐쇄하고 다시 입금한다.
- E14. 현금 보유량이 부족할 경우 알림을 발생시킨다.
- E15. 명세표 출력 용지가 부족할 경우 알림을 발생시킨다.

2040

**Actors**

- E1. 본인확인 실패 시, 종료한다.
- E5. 계좌 잔액 부족 시, 종료한다.
- E14. 현금 보유량이 부족할 경우 알림을 발생시킨다.
- E15. 명세표 출력 용지가 부족할 경우 알림을 발생시킨다.

→ 2030 과 2040, 2041 Use Case 불일치



## 2031. Describe Use Case

## 1.4 Transfer

2030

Alternative Courses  
of Events

E0. 한국 계좌의 경우 한국어 모드, 해외 계좌일 경우 외국어 모드  
E12. 명세표 출력 시, 거래 내역 출력 / 미출력시 다음 단계.

2040

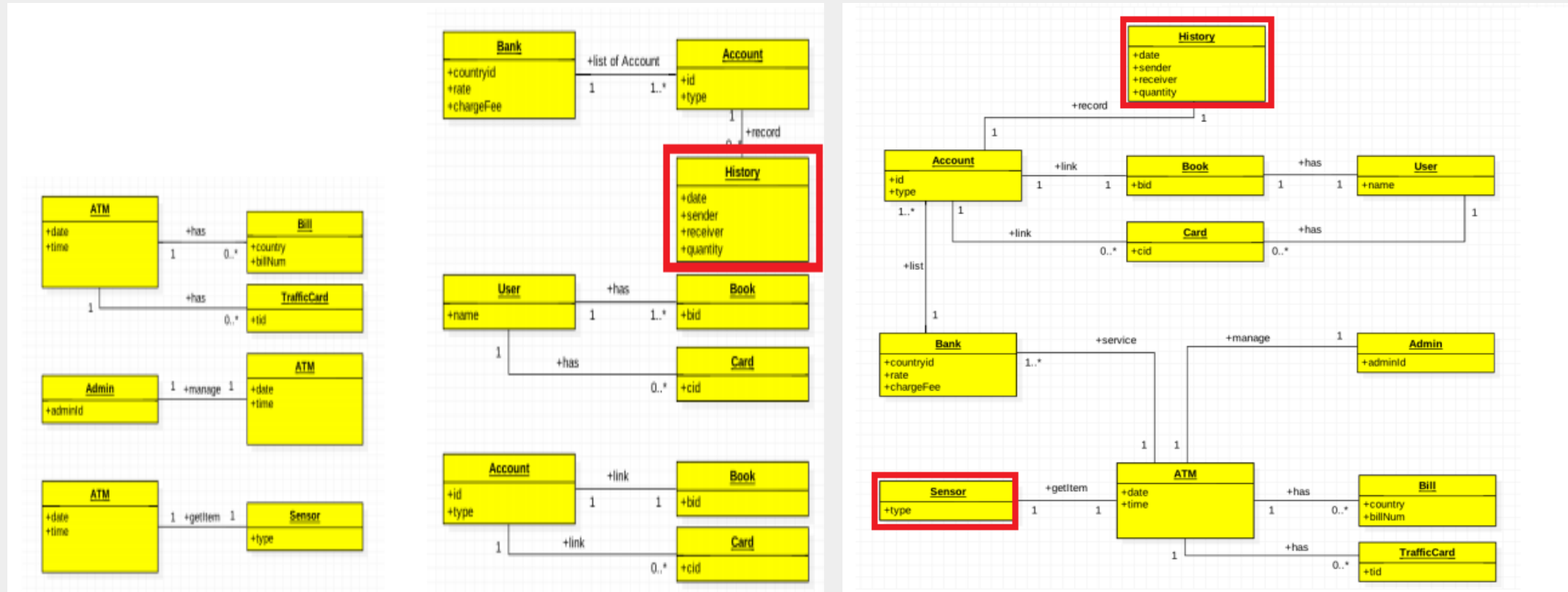
Alternative Courses  
of Events

E0. 한국 계좌의 경우 한국어 모드, 해외 계좌일 경우 외국어 모드  
E0. 다른 거래 선택  
E12. 명세표 출력 시, 거래 내역 출력 / 미출력시 다음 단계.

→ 2030 과 2040, 2041 Use Case 불일치

다른 Use Case도 모두 다르므로 생략

## 2033. Define Domain Model



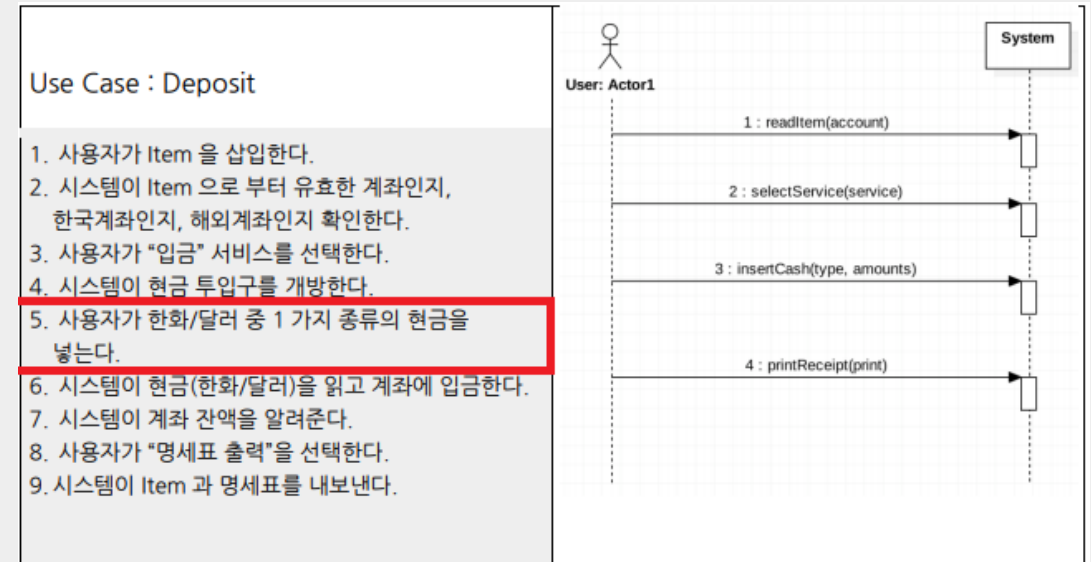
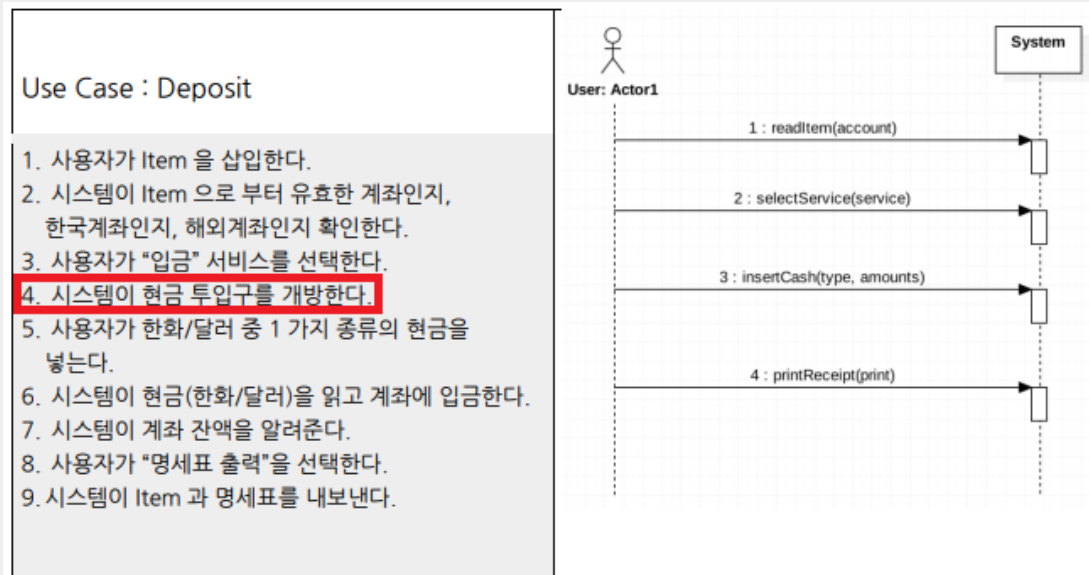
→ 사용하지 않는 domain들 정의

## 2034. Refine Clossary

Check	Use cases	계좌 조회
Deposit	Use cases	예금 입금
Withdraw	Use cases	예금 인출
Transfer	Use cases	예금 송금
Issue TrafficCard	Use cases	교통카드 발급
<b>Update</b>	Use cases	거래 내역 연동
<b>Verify Sufficient Fund</b>	Use cases	충분한 잔액 확인
<b>Status Alarm</b>	Use cases	ATM 현금/교통카드 보유량 부적절 경고 알림
<b>Management</b>	Use cases	ATM 관리자 현금/교통카드 보유량 관리
<b>Bank</b>	Concept	ATM System 에 등록 되어있는 은행
<b>countryid</b>	Attribute	은행 국가(한/미)
<b>Rate</b>	Attribute	환율
<b>chargeFee</b>	Attribute	수수료
<b>Account</b>	Concept	계좌
<b>id</b>	Attribute	계좌 식별번호
<b>type</b>	Attribute	계좌 종류
<b>History</b>	Concept	거래 내역
<b>date</b>	Attribute	거래 날짜/시간
<b>sender</b>	Attribute	송신 계좌

→사용되지 않은 변수

2035. Sequence Diagram



→ 이전 발표내용 반영되지 않음

## 2036. Define Operation Contracts

## 6.2 selectService(service)

Post-conditions	비밀번호를 확인한다.
-----------------	-------------

→ 모든 Service가 비밀번호를 필요로 하지는 않으므로 수정 필요.

ex) Deposit

## 2036. Define Operation Contracts

## 6.5 insertCash(type, amounts)

Name	insertCash(type, amount)
Responsibilities	입금한 지폐의 종류와 양을 확인한다.
Type	System
Cross References	Functions : R.1.1 Use cases : Deposit
Notes	
Exceptions	-유효한 지폐(원화/달러)가 아닐경우
Output	입금한 지폐의 종류와 양을 User 에게 다시 확인
Pre-conditions	입금 서비스가 선택 되어있어야 한다
Post-conditions	명세표 확인 및 출력을 한다.

→ Exceptions에 '원화와 달러를 섞어서 입금할 경우 입금하지 않는다' 추가

## 2036. Define Operation Contracts

### 6.11 printReceipt(print)

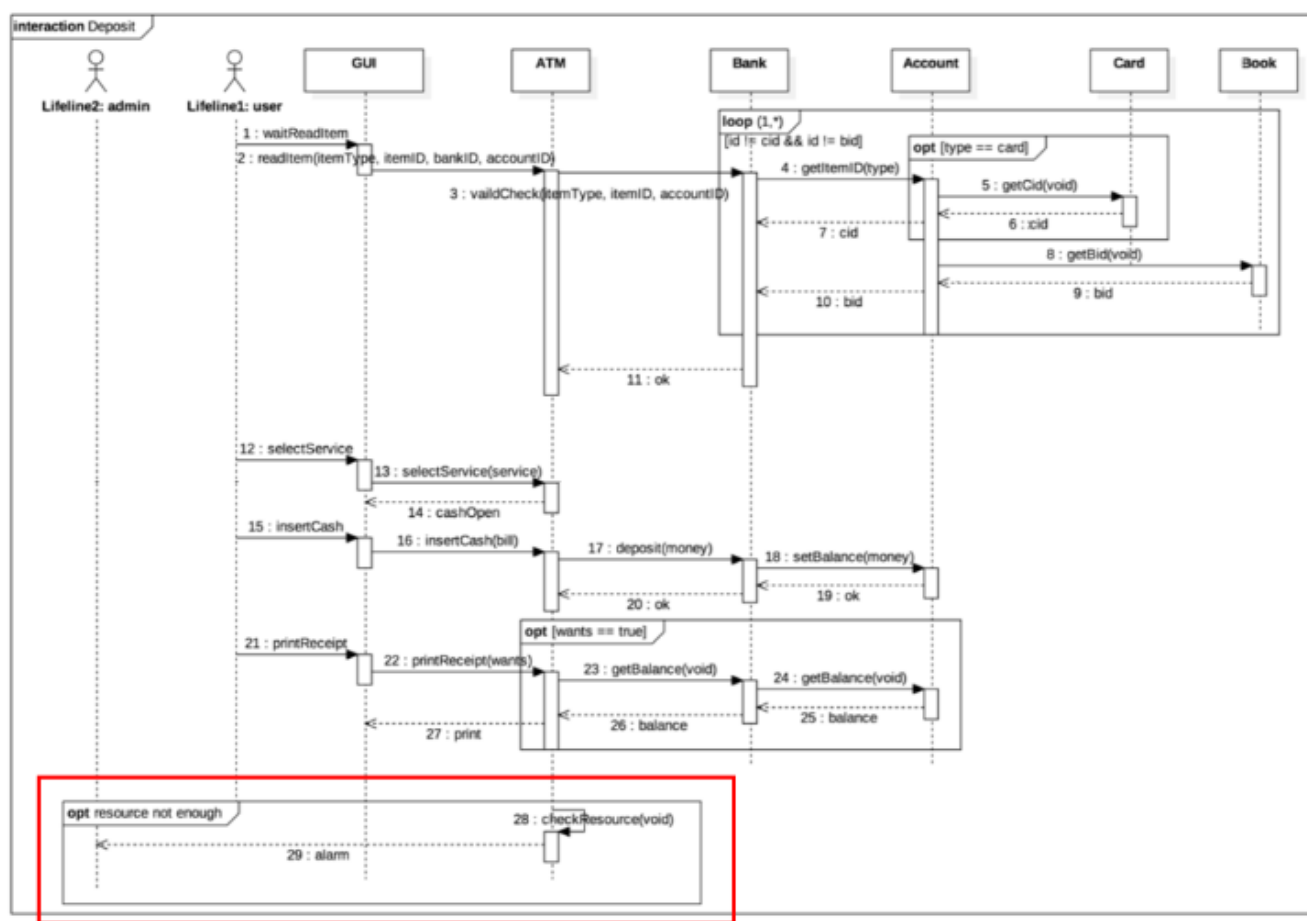
→ 6.3의 printReceipt와 중복되므로 삭제 필요

## 2144. Define Interaction Diagrams

- 이전 발표 이후로 **반영된 사항 없음**
- 이전 발표에서 문제되었던 부분 확인 요망합니다…….



## 2144. Define Interaction Diagrams



명세에 언급되어 있던 end가  
interaction diagram 어디에도 존재하지 않음.

Update관련 존재하지 않음.

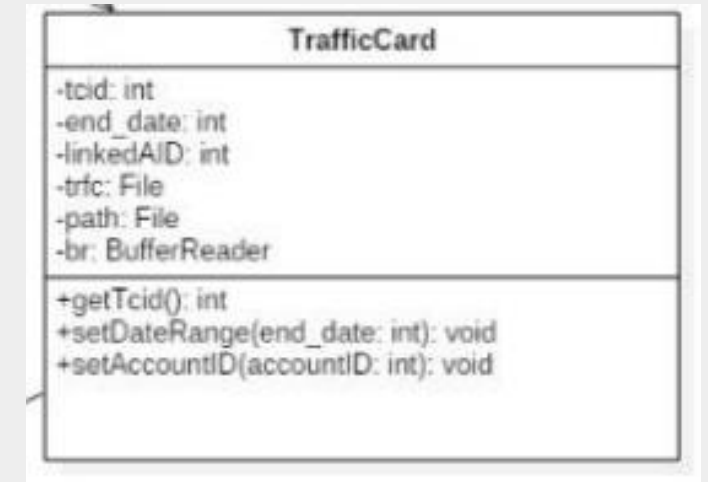
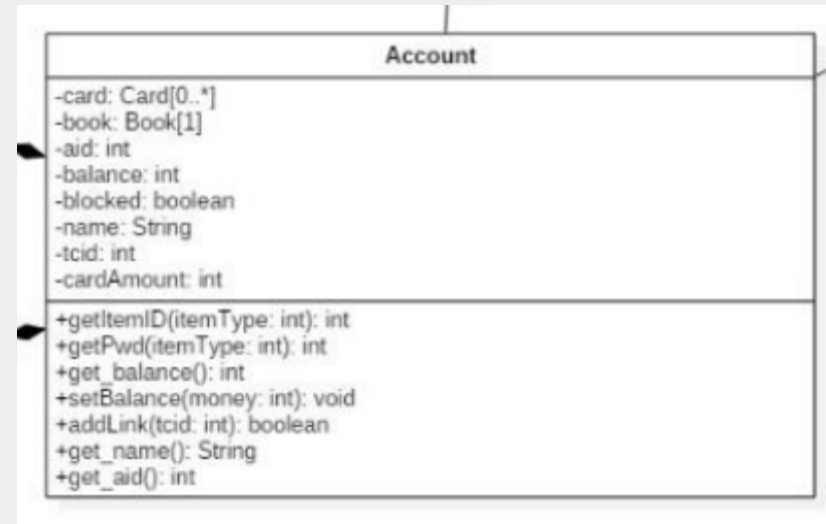
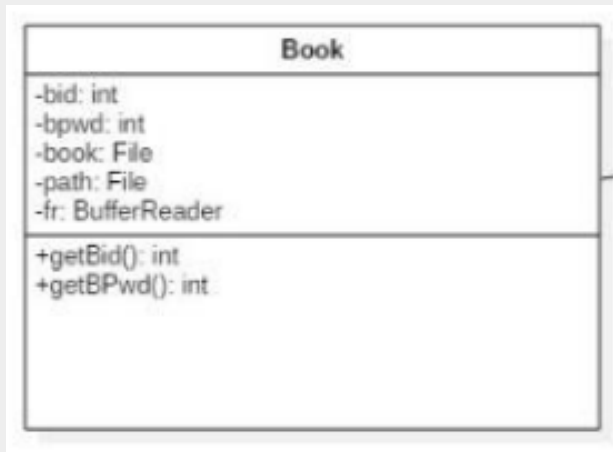
## 2045. Define Design Class Diagrams



- ✓ 2044에 사용한 함수들이 명시되어 있지 않다.
- ✓ cashAmount의 데이터형이 다르다.
- ✓ ATMadminID, ATMnation 변수가 누락되어 있다.
- ✓ confirmedItem, PWD는 프로그램상에서 존재하지 않는 변수이다.
- ✓ 함수의 이름이 다르다. (setDateRange→ setDataRange)
- ✓ 함수의 반환값이 다르다. (destAccount의 반환값은 string이다.)
- ✓ checkResource의 접근제어지시자는 public이다.
- ✓ bankDataDownload와 bootATM은 존재하지 않는다.
- ✓ 중복된 변수가 존재한다.



## 2045. Define Design Class Diagrams



- ✓ Book : 코드에 사용되는 File Reader 가 빠져있음
- ✓ Account : 코드에 존재하지 않는 blocked 변수 존재, book은 배열이 아님
- ✓ TrafficCard : 코드에 사용되는 File Reader & Count 존재하지 않음

## 2051. Implement Class & Method

- ✓ 모든 Class와 Method의 Cross Reference가 ctrl+c , ctrl+v...
- ✓ 적절하지 않은 Cross Reference 기재 → 이하 생략

## 2051. Implement Class &amp; Method

## 1.1.5 insertCash

Type	Method
Name	insertCash
Purpose	User 가 입금하려는 금액을 넣는다.
Cross Reference	System Function : R.1.0 , R.1.1 , R.1.2 , R.1.3 , R.1.4 , R.2.0 , R.2.1 , R.2.2, R.3.0 Use cases : "Check" , "Deposit" , "Transfer" , "Withdraw" , "Management" , "Update" , "Verify Sufficient Fund" , "Status Alarm"
Input	String[] bill
Output	int
Abstract operation	지폐 code 배열을 받고 그에 알맞은 돈을 bank 에 입금해준다.
Exceptional Course of Events	ATM 기기안 현금이 너무 많으면 돈을 더 못 넣으니 return false 를 하게 된다.

→ 존재하지 않는 Update, '너무 많다'의 기준이 명확하지 않음

## 2051. Implement Class &amp; Method

## 1.1.8 printReceipt &amp; setDateRange

Type	Method
Name	printReceipt
Purpose	명세표를 출력한다.
Cross Reference	System Function : R.1.0 , R.1.1 , R.1.2 , R.1.3 , R.1.4 , R.2.0 , R.2.1 , R.2.2 , R.3.0 Use cases : "Check" , "Deposit" , "Transfer" , "Withdraw" , "Management" , "Update" , "Verify Sufficient Fund" , "Status Alarm"
Input	
Output	boolean
Abstract operation	Parameter wants == true 이면 , balance 를 bank 로부터 받아온다.
Exceptional Course of Events	

→ 넘버링 오류 (번호 겹침)

Type	Method
Name	setDateRange
Purpose	교통카드 이용 날짜를 설정한다.

→ 오타 (setDataRange), 코드상에서 사용되지 않은 메소드

## 2051. Implement Class &amp; Method

1.1.11 end

1.1.11. end

Type	Method
Name	end
Purpose	관리자가 작업을 끝내면 다시 파일을 읽어 atm 정보를 업데이트 한다.
Cross Reference	System Function : R.1.0 , R.1.1 , R.1.2 , R.1.3 , R.1.4 , R.2.0 , R.2.1 , R.2.2 , R.3.0 Use cases : "Check" , "Deposit" , "Transfer" , "Withdraw" , "Management" , "Update" , "Verify Sufficient Fund" , "Status Alarm"
Input	
Output	Void
Abstract operation	관리자가 작업을 끝내면 다시 파일을 읽어 atm 정보를 업데이트 한다.
Exceptional Course of Events	-

→ 이전 단계(1000, 2030) 에서 나오지 않음.



## 2051. Implement Class &amp; Method

## 1.2.2 Confirm &amp; WriteData

1.2.2. Confirm

Type	Method
Name	Confirm
Purpose	입력한 비밀번호가 맞는지 확인한다.
Cross Reference	System Function : R.1.0 , R.1.1 , R.1.2 , R.1.3 , R.1.4 , R.2.0 , R.2.1 , R.2.2 , R.3.0 Use cases : "Check" , "Deposit" , "Transfer" , "Withdraw" , "Management" , "Update" , "Verify Sufficient Fund" , "Status Alarm"
Input (Method)	<code>int_pwd</code>
Output (Method)	<code>boolean</code>
Abstract operation (Method)	item cache 인 Type 변수를 통해 카드인지 통장인지 구분한다.
Exceptional Course of Events	불러온 계좌와 해당 비밀번호가 일치하지 않으면 false 를 반환한다.

→ item cache인 Type 변수..?

1.2.2. WriteData

Type	Method
Name	WriteData
Purpose	계좌파일에 바뀐 정보를 쓴다 (파일 데이터 쓰기)
Cross Reference	System Function : R.1.0 , R.1.1 , R.1.2 , R.1.3 , R.1.4 , R.2.0 , R.2.1 , R.2.2 , R.3.0 Use cases : "Check" , "Deposit" , "Transfer" , "Withdraw" , "Management" , "Update" , "Verify Sufficient Fund" , "Status Alarm"
Input (Method)	
Output (Method)	
Abstract operation (Method)	
Exceptional Course of Events	-

→ 넘버링 오류 (번호 겹침)

## 2051. Implement Class & Method

### 1.2.3 getBalance & ValidCheck

→ 넘버링 오류 (번호 겹침)

## 2051. Implement Class &amp; Method

## 1.2.5 linkAccount

1.2.5. linkAccount	
Type	Method
Name	linkAccount
Purpose	교통카드와 계좌를 연동시킨다.
Cross Reference	System Function : R.1.0 , R.1.1 , R.1.2 , R.1.3 , R.1.4 , R.2.0 , R.2.1 , R.2.2 , R.3.0 Use cases : "Check" , "Deposit" , "Transfer" , "Withdraw" , "Management" , "Update" , "Verify Sufficient Fund" , "Status Alarm"
Input (Method)	int_tcid
Output (Method)	boolean
Abstract operation (Method)	
Exceptional Course of Events	-현재 계좌가 유효하지 않으면 false 를 반환한다.

→ Cross Reference 누락

## 2051. Implement Class &amp; Method

## 1.2.6 transfer

Type	Method
Name	transfer
Purpose	송금한다.
Cross Reference	System Function : R.1.0 , R.1.1 , R.1.2 , R.1.3 , R.1.4 , R.2.0 , R.2.1 , R.2.2 , R.3.0 Use cases : "Check" , "Deposit" , "Transfer" , "Withdraw" "Management" , "Update" , "Verify Sufficient Fund" , "Status Alarm"
Input (Method)	int_money
Output (Method)	boolean
Abstract operation (Method)	같은 은행이라면 파일에 데이터를 두번 덮어쓰지 않도록 한 객체 데이터를 변경한다.
Exceptional Course of Events	내 계좌의 잔액이 부족하거나 유효하지 않으면 false 를 반환한다.

→ Cross Reference 누락

## 2051. Implement Class &amp; Method

## 1.3.4 set\_balance

Type	Method
Name	set_balance
Purpose	해당 계좌의 잔고를 증감한다.
Cross Reference	System Function : R.1.0 , R.1.1 , R.1.2 , R.1.3 , R.1.4 , R.2.0 , R.2.1 , R.2.2 , R.3.0 Use cases : "Check" , "Deposit" , "Transfer" , "Withdraw" , "Management" , "Update" , "Verify Sufficient Fund" , "Status Alarm"
Input (Method)	
Output (Method)	
Abstract operation (Method)	
Exceptional Course of Events	-

```
// set balance
public void set_balance(int _bal) {
    this.balance = _bal;
}
```

→ 증감하지 않는다.

## 2051. Implement Class &amp; Method

## 2.1 waitReadItem

Name	waitReadItem
Responsibilities	User 가 Item 을 인식시키는 것을 기다린다.
Type	GUI
Cross Reference	System Function : R.1.0 , R.1.1 , R.1.2 , R.1.3 , R.1.4 , R.2.0 , R.2.1 , R.2.2, R.3.0 Use cases : "Check" , "Deposit" , "Transfer" , "Withdraw" , "Management" , "Update" , "Verify Sufficient Fund" , "Status Alarm"
Notes	-
PreConditions	ATM 실행 상태
PostConditions	User 의 Item 값 입력

→ 기다리는 상황 x

## 2051. Implement Class &amp; Method

## 2.5 selectNation

Name	select <b>Natinon</b>
Responsibilities	출금하고자하는 지폐 종류를 선택한다.
Type	GUI
Cross Reference	System Function : R.1.3 Use cases : "Withdraw"
Notes	-
Pre-Conditions	출금 서비스 선택
Post-Conditions	원 / 달러 선택

→ Natinon 오타 수정 필요

# 02. System Testing



Categorize

Group	Category	Values	#
Read Item	Read Item	Valid Input	100
		Invalid Item Type	101
		Invalid Item Id	102
		Invalid Item Bank	103
		Invalid Item Account ID	104
Mode	Mode	Check	200
		Deposit	201
		Withdraw	202
		Transfer	203
		Issue Traffic Card	204
Password	Password	Valid Password	300
		Invalid Password	301
Data	Select Data	Valid Money Once	400
		Valid Single Type Money	401
		Valid Multiple Type Money	402
		Invalid Money	403
		Currency	404
	Select Bank	Bank.	410
	Select Account ID	Valid Account ID	420
		Invalid Account ID	421
	Input Data	Valid Money	430
		Invalid Money	431
		Valid Duration	432
		Invalid Duration	433

## Categorize

<b>Agreement</b>	Agreement	Agree	500
		Disagree	501
<b>Change Balance</b>	Actor	Increase Balance	600
		Decrease Balance	601
		Send Money	602
	Admin	Increase Balance	610
		Decrease Balance	611
<b>Print Receipt</b>	Receipt	Print Receipt	700
		Not Print Receipt	701
<b>Management</b>	Manage TC	Not Enough TC	800
		Enough TC	801
	Manage Receipt	Not Enough Receipt	810
		Enough Receipt	811
	Manage Money	Not Enough Money	820
		Enough Money	821
		Too Many Money	822
<b>Alarm</b>	Alarm	Send Alarm	900
		Not Send Alarm	901
<b>Cancel</b>	Cancel	Cancel	1000
		Clear	1001
		Backspace	1002

## Description

Values	Description
Valid Input	4가지 항목에 모두 올바른 값을 입력한 경우
Invalid Item Type	잘못된 Item Type을 입력한 경우
Invalid Item Id	잘못된 Item ID를 입력한 경우
Invalid Item Bank	잘못된 Item Bank를 입력한 경우
Invalid Item Account ID	잘못된 Item Account ID를 입력한 경우
Check	계좌의 잔액을 확인한다
Deposit	돈을 예금한다.
Withdraw	돈을 인출한다.
Transfer	돈을 송금한다.
Issue Traffic Card	교통카드를 발급한다.
Valid Password	유효한 비밀번호를 입력한.
Invalid Password	유효하지 않은 비밀번호를 입력한다.
Valid Money Once	금액을 한번 선택한다.
Valid Single Type Money	한 종류의 금액을 여러 번 선택한다.
Valid Multiple Type Money	여러 종류의 금액을 여러 번 선택한다.
Invalid Money	유효하지 않은 금액을 입력한다.
Currency	거래를 진행할 화폐를 선택한다.
Bank.	거래를 진행할 은행을 선택한다.
Valid Account ID	유효한 Account ID를 입력한다.
Invalid Account ID	유효하지 않은 Account ID를 입력한다.
Valid Money	유효한 금액을 입력한다.
Invalid Money	유효하지 않은 금액을 입력한다.

## Description

Valid Duration	유효한 기간을 입력한다.
Invalid Duration	유효하지 않은 기간을 입력한다.
Agree	이용 약관에 동의한다.
Disagree	이용 약관에 동의하지 않는다.
Increase Balance	행위자의 금액을 증가시킨다.
Decrease Balance	행위자의 금액을 감소시킨다.
Send Money	돈을 송금한다.
Increase Balance	관리자의 금액을 증가시킨다.
Decrease Balance	관리자의 금액을 감소시킨다.
Print Receipt	영수증을 출력한다.
Not Print Receipt	영수증을 출력하지 않는다.
Not Enough TC	교통카드가 부족하다.
Enough TC	교통카드가 충분하다.
Not Enough Receipt	영수증을 출력할 용지가 부족하다.
Enough Receipt	영수증을 출력할 용지가 충분하다.
Not Enough Money	ATM의 지폐가 충분하지 않다.
Enough Money	ATM의 지폐가 충분하다.
Too Many Money	ATM의 지폐가 너무 많다.
Send Alarm	관리자에게 알람을 보낸다.
Not Send Alarm	관리자에게 알람을 보내지 않는다.
Cancel	선택을 취소한다.
Clear	선택을 초기화한다.
Backspace	메인화면으로 돌아간다.

```

# Read Item
Read Item:
    Valid Input,
    Invalid ItemType,
    Invalid ItemId,
    Invalid ItemBank,
    Invalid ItemAccountId,
    [error]
    [error]
    [error]
    [error]

# Mode
Mode:
    Check,
    Deposit,
    Withdraw,
    Transfer,
    Issue TrafficCard,
    [property Check]
    [property Deposit]
    [property Withdraw]
    [property Transfer]
    [property IssueTC]

# Password
Password:
    Valid Password,
    Invalid Password,
    [if Check || Withdraw || Transfer || IssueTC]
    [error] [if Check || Withdraw || Transfer || IssueTC]

# Data
Select Data:
    Valid Money Once,
    Valid Single Type Money,
    Valid Multiple Type Money,
    Invalid Money,
    Currency,
    [if Deposit]
    [if Deposit]
    [if Deposit]
    [error] [if Deposit] #Overflow
    [if Withdraw]

Select Bank:
    Bank,
    [if Transfer]

Select AccountID:
    Valid AccountID,
    Invalid AccountID,
    [if Transfer]
    [error] [if Transfer] #존재하지 않는 AccountID 입력

Input Data:
    Valid Money,
    Invalid Money,
    Valid Duration,
    Invalid Duration,
    [if Withdraw || Transfer]
    [error] [if Withdraw || Transfer] #가진 금액 이상 입력
    [if IssueTC]
    [error] [if IssueTC] #00일

# Agreement
Agreement:
    Agree,
    Disagree,
    [if Transfer]
    [if Transfer]

```

```

# Change Balance
Actor:
    Increase Balance,           [if Deposit]
    Decrease Balance,         [if Withdraw || IssueTC]
    Send Money,               [if Transfer]

Admin:
    Increase Balance,         [if Deposit || IssueTC] [property IncreaseBalance]
    Decrease Balance,         [if Transfer || Withdraw] [property DecreaseBalance]

# Print Receipt
Receipt:
    Print Receipt,           [property PrintReceipt]
    Not Print Receipt,

# Management
Manage TC:
    Not Enough TC,           [if IssueTC] [property TCAlarm]
    Enough TC,               [if IssueTC]

Manage Receipt:
    Not Enough Receipt,      [if PrintReceipt][property ReceiptAlarm]
    Enough Receipt,          [if PrintReceipt]

Manage Money:
    Not Enough Money,        [if DecreaseBalance][property NMoneyAlarm]
    Enough Money,            [if IncreaseBalance || DecreaseBalance]
    Too Many Money,         [if IncreaseBalance][property TMoneyAlarm]

# Alarm
Alarm:
    Send Alarm,              [if TCAlarm || ReceiptAlarm || NMoneyAlarm || TMoneyAlarm]
    Not Send Alarm,         [error]

# Cancel
Cancel:
    Cancel,                  [single]
    Clear,                   [single]
    Backspace,               [single]

```

## SV팀의 에러 수정

```
"C:\Program Files\Java\jre-10.0.1\bin\java" "-javaagent:C:\Program Files\JetBrains\Inte
java.io.FileNotFoundException: TrafficCard.txt (지정된 파일을 찾을 수 없습니다)
  at java.base/java.io.FileInputStream.open0(Native Method)
  at java.base/java.io.FileInputStream.open(Unknown Source)
  at java.base/java.io.FileInputStream.<init>(Unknown Source)
  at java.base/java.io.FileReader.<init>(Unknown Source)
  at Item.TrafficCard.<init>(TrafficCard.java:43)
  at ATM.ATM.<init>(ATM.java:22)
  at GUI.GUIController.readyReadCard(GUIController.java:118)
  at GUI.GUIController.<init>(GUIController.java:110)
  at GUI.GUIController.main(GUIController.java:103)
java.io.FileNotFoundException: management.txt (지정된 파일을 찾을 수 없습니다)
  at java.base/java.io.FileInputStream.open0(Native Method)
  at java.base/java.io.FileInputStream.open(Unknown Source)
  at java.base/java.io.FileInputStream.<init>(Unknown Source)
  at java.base/java.io.FileReader.<init>(Unknown Source)
  at ATM.ATM.<init>(ATM.java:60)
  at GUI.GUIController.readyReadCard(GUIController.java:118)
  at GUI.GUIController.<init>(GUIController.java:110)
  at GUI.GUIController.main(GUIController.java:103)
Hi..!
java.io.FileNotFoundException: citi.txt (지정된 파일을 찾을 수 없습니다)
  at java.base/java.io.FileInputStream.open0(Native Method)
  at java.base/java.io.FileInputStream.open(Unknown Source)
  at java.base/java.io.FileInputStream.<init>(Unknown Source)
  at BankSystem.Bank.loadData(Bank.java:44)
```

수정해준 에러를 복구한 SMA팀

TestCase	Result
Test1- 101	Failed
Test2- 102	Failed
Test3- 103	Failed
Test4- 104	Failed
Test5- 301	Failed
Test6- 403	Failed
Test7- 421	Failed
Test8- 431	Failed
Test9- 433	Failed
Test10- 901	Failed
Test11- 1000	Failed
Test12- 1001	Failed
Test13- 1002	Failed
Test14- 100.200.300.700.810.900	Failed
Test15- 100.200.300.700.811	Failed
Test16- 100.200.300.701	Failed
Test17- 100.201.400.600.610.700.810.821.900	Failed
Test18- 100.201.400.600.610.700.810.822.900	Failed
Test19- 100.201.400.600.610.700.811.821	Failed
Test20- 100.201.400.600.610.700.811.822.900	Failed
Test21- 100.201.400.600.610.701.821	Failed
Test22- 100.201.400.600.610.701.822.900	Failed
Test23- 100.201.401.600.610.700.810.821.900	Failed
Test24- 100.201.401.600.610.700.810.822.900	Failed
Test25- 100.201.401.600.610.700.811.821	Failed
Test26- 100.201.401.600.610.700.811.822.900	Failed
Test27- 100.201.401.600.610.701.821	Failed
Test28- 100.201.401.600.610.701.822.900	Failed
Test29- 100.201.402.600.610.700.810.821.900	Failed
Test30- 100.201.402.600.610.700.810.822.900	Failed
Test31- 100.201.402.600.610.700.811.821	Failed
Test32- 100.201.402.600.610.700.811.822.900	Failed
Test33- 100.201.402.600.610.701.821	Failed
Test34- 100.201.402.600.610.701.822.900	Failed
Test35- 100.202.300.404.430.601.611.700.810.820.900	Failed

0% Passed



0% Passed

Test36- 100.202.300.404.430.601.611.700.810.821.900	Failed
Test37- 100.202.300.404.430.601.611.700.811.820.900	Failed
Test38- 100.202.300.404.430.601.611.700.811.821	Failed
Test39- 100.202.300.404.430.601.611.701.820.900	Failed
Test40- 100.202.300.404.430.601.611.701.821	Failed
Test41- 100.203.300.410.420.430.500.602.611.700.810.820.900	Failed
Test42- 100.203.300.410.420.430.500.602.611.700.810.821.900	Failed
Test43- 100.203.300.410.420.430.500.602.611.700.811.820.900	Failed
Test44- 100.203.300.410.420.430.500.602.611.700.811.821	Failed
Test45- 100.203.300.410.420.430.500.602.611.701.820.900	Failed
Test46- 100.203.300.410.420.430.500.602.611.701.821	Failed
Test47- 100.203.300.410.420.430.501.602.611.700.810.820.900	Failed
Test48- 100.203.300.410.420.430.501.602.611.700.810.821.900	Failed
Test49- 100.203.300.410.420.430.501.602.611.700.811.820.900	Failed
Test50- 100.203.300.410.420.430.501.602.611.700.811.821	Failed
Test51- 100.203.300.410.420.430.501.602.611.701.820.900	Failed
Test52- 100.203.300.410.420.430.501.602.611.701.821	Failed
Test53- 100.204.300.432.601.610.700.801.810.821.900	Failed
Test54- 100.204.300.432.601.610.700.801.810.822.900	Failed
Test55- 100.204.300.432.601.610.700.801.811.821.900	Failed
Test56- 100.204.300.432.601.610.700.801.811.822.900	Failed
Test57- 100.204.300.432.601.610.700.801.810.821.900	Failed
Test58- 100.204.300.432.601.610.700.801.810.822.900	Failed
Test59- 100.204.300.432.601.610.700.801.811.821	Failed
Test60- 100.204.300.432.601.610.700.801.811.822.900	Failed
Test61- 100.204.300.432.600.610.701.801.821.900	Failed
Test62- 100.204.300.432.601.610.701.801.822.900	Failed
Test63- 100.204.300.432.601.610.701.801.821	Failed
Test64- 100.204.300.432.601.610.701.801.822.900	Failed
전체 Test Case: 64개	Pass: 0개

25% Passed

TestCase	Result(2nd)	Fail Reason (2nd)
Test1- 101	Passed	
Test2- 102	Passed	
Test3- 103	Passed	
Test4- 104	Passed	
Test5- 301	Passed	
Test6- 403	Passed	
Test7- 421	Passed	
Test8- 431	Passed	
Test9- 433	Failed	교통카드 개수가 남아있음에도 교통카드 개수가 부족하다고 함
Test10- 901	Failed	Alarm을 보내지 않는 경우는 Error 상황
Test11- 1000	Passed	
Test12- 1001	Passed	
Test13- 1002	Passed	
Test14- 100.200.300.700.810.900	Failed	영수증이 부족한 경우에 알람을 보내지 않음
Test15- 100.200.300.700.811	Passed	
Test16- 100.200.300.701	Passed	
Test17- 100.201.400.600.610.700.810.821.900	Failed	영수증이 부족한 경우에 알람을 보내지 않음
Test18- 100.201.400.600.610.700.810.822.900	Failed	영수증이 부족하고 돈이 너무 많은 경우 알람을 보내지 않음
Test19- 100.201.400.600.610.700.811.821	Failed	50000원을 넣었을 때 화면에 5000원이 출력된다.
Test20- 100.201.400.600.610.700.811.822.900	Failed	돈이 너무 많은 경우 알람을 보내지 않음
Test21- 100.201.400.600.610.701.821	Failed	50000원을 넣었을 때 화면에 5000원이 출력된다.
Test22- 100.201.400.600.610.701.822.900	Failed	돈이 너무 많은 경우 알람을 보내지 않음
Test23- 100.201.401.600.610.700.810.821.900	Failed	영수증이 부족한 경우 알람을 보내지 않음
Test24- 100.201.401.600.610.700.810.822.900	Failed	영수증이 부족하거나 돈이 너무 많은 경우 알람을 보내지 않음
Test25- 100.201.401.600.610.700.811.821	Failed	50000원을 넣었을 때 화면에 5000원이 출력된다.
Test26- 100.201.401.600.610.700.811.822.900	Failed	돈이 너무 많은 경우 알람을 보내지 않음
Test27- 100.201.401.600.610.701.821	Failed	50000원을 넣었을 때 화면에 5000원이 출력된다.
Test28- 100.201.401.600.610.701.822.900	Failed	돈이 너무 많은 경우 알람을 보내지 않음
Test29- 100.201.402.600.610.700.810.821.900	Failed	여러 화폐가 동시에 입금되지 않으며 영수증이 부족한 경우 알람을 보내지 않음
Test30- 100.201.402.600.610.700.810.822.900	Failed	여러 화폐가 동시에 입금되지 않으며 영수증이 부족하거나 돈이 많은 경우 알람을 보내지 않음
Test31- 100.201.402.600.610.700.811.821	Failed	50000원을 넣었을 때 화면에 5000원이 출력된다.
Test32- 100.201.402.600.610.700.811.822.900	Failed	여러 화폐가 동시에 입금되지 않으며 돈이 너무 많은 경우 알람을 보내지 않음
Test33- 100.201.402.600.610.701.821	Failed	50000원을 넣었을 때 화면에 5000원이 출력된다.
Test34- 100.201.402.600.610.701.822.900	Failed	여러 화폐가 동시에 입금되지 않으며 돈이 너무 많은 경우 알람을 보내지 않음.
Test35- 100.202.300.404.430.601.611.700.810.820.900	Failed	영수증이 부족하거나 돈이 부족한 경우 알람을 보내지 않음
Test36- 100.202.300.404.430.601.611.700.810.821.900	Failed	영수증이 부족한 경우 알람을 보내지 않음
Test37- 100.202.300.404.430.601.611.700.811.820.900	Failed	돈이 부족한 경우 알람을 보내지 않음
Test38- 100.202.300.404.430.601.611.700.811.821	Failed	돈을 0원 또는 0달러 출금시 수수료만 요구
Test39- 100.202.300.404.430.601.611.701.820.900	Failed	돈이 부족한 경우 알람을 보내지 않음
Test40- 100.202.300.404.430.601.611.701.821	Failed	돈을 0원 또는 0달러 출금시 수수료만 요구
Test41- 100.203.300.410.420.430.500.602.611.700.810.820.9	Failed	영수증이 부족하거나 돈이 부족한 경우 알람을 보내지 않음

25% Passed

Test42- 100.203.300.410.420.430.500.602.611.700.810.821.9	Failed	영수증이 부족한 경우 알람을 보내지 않음
Test43- 100.203.300.410.420.430.500.602.611.700.811.820.9	Failed	돈이 부족한 경우 알람을 보내지 않음
Test44- 100.203.300.410.420.430.500.602.611.700.811.821	Passed	
Test45- 100.203.300.410.420.430.500.602.611.701.820.900	Failed	돈이 부족한 경우 알람을 보내지 않음
Test46- 100.203.300.410.420.430.500.602.611.701.821	Passed	
Test47- 100.203.300.410.420.430.501.602.611.700.810.820.9	Failed	영수증이 부족하거나 돈이 부족한 경우 알람을 보내지 않음
Test48- 100.203.300.410.420.430.501.602.611.700.810.821.9	Failed	영수증이 부족한 경우 알람을 보내지 않음
Test49- 100.203.300.410.420.430.501.602.611.700.811.820.9	Failed	돈이 부족한 경우 알람을 보내지 않음
Test50- 100.203.300.410.420.430.501.602.611.700.811.821	Failed	한 자리수(예를 들어 2)를 입력하기 위해 2 Enter를 누르면 제대로 인식되지 않고 002를 눌러야 함
Test51- 100.203.300.410.420.430.501.602.611.701.820.900	Failed	돈이 부족한 경우 알람을 보내지 않음
Test52- 100.203.300.410.420.430.501.602.611.701.821	Failed	한 자리수(예를 들어 2)를 입력하기 위해 2 Enter를 누르면 제대로 인식되지 않고 002를 눌러야 함
Test53- 100.204.300.432.601.610.700.801.810.821.900	Failed	교통카드가 부족하거나 영수증이 부족한 경우 알람을 보내지 않음
Test54- 100.204.300.432.601.610.700.801.810.822.900	Failed	교통카드가 부족하거나 영수증이 부족하거나 돈이 많은 경우 알람을 보내지 않음
Test55- 100.204.300.432.601.610.700.801.811.821.900	Failed	교통카드가 부족한 경우 알람을 보내지 않음
Test56- 100.204.300.432.601.610.700.801.811.822.900	Failed	교통카드가 부족하거나 돈이 많은 경우 알람을 보내지 않음
Test57- 100.204.300.432.601.610.700.801.810.821.900	Failed	영수증이 부족한 경우 알람을 보내지 않음
Test58- 100.204.300.432.601.610.700.801.810.822.900	Failed	영수증이 부족하거나 돈이 많은 경우 알람을 보내지 않음
Test59- 100.204.300.432.601.610.700.801.811.821	Failed	문서에는 카드발급수수료가 0.3달러라고 되어있으나 3달러가 빠져나감
Test60- 100.204.300.432.601.610.700.801.811.822.900	Failed	돈이 너무 많은 경우 알람을 보내지 않음
Test61- 100.204.300.432.600.610.701.801.821.900	Failed	교통카드가 부족한 경우 알람을 보내지 않음
Test62- 100.204.300.432.601.610.701.801.822.900	Failed	교통카드가 부족하고 돈이 많은 경우 알람을 보내지 않음
Test63- 100.204.300.432.601.610.701.801.821	Failed	문서에는 카드발급수수료가 0.3달러라고 되어있으나 3달러가 빠져나감
Test64- 100.204.300.432.601.610.701.801.822.900	Failed	돈이 너무 많은 경우 알람을 보내지 않음
전체 Test Case: 64개	Pass: 15개	Pass Percentage: 25%

- ✓ 1<sup>st</sup> Category Partition Testing => 42% Passed
- ✓ 2<sup>nd</sup> Category Partition Testing => 25% Passed

=> 첫번째 CPT보다 낮아진 Passed Percentage

- File 경로, Parsing 中 Array Index Out of Bounds와 같은 치명적인 Error 수정 실패.
- Admin에게 알림을 주지 않음
- 몇가지 예외처리가 잘 되어있지 않음
- 수수료 등 추가된 일부 기능에서 에러 발생

## 결과 파일

ReadItem	Mode	InputPassword	InputBank	InputAccountID	InputDuration	InputCurrency	InputMoney	Agreement	TCArm	ActorTransaction	AdminTransaction			
Valid	Check	Valid	Skip	Skip	Skip	Skip	Skip	NotPrint	Send					
Valid	Check	Valid	Skip	Skip	Skip	Skip	Skip	Print	Send					
Valid	Deposit	Skip	Skip	Skip	Skip	Skip	Skip	IncreaseBalance	IncreaseBalance	Send	Print	Send		
Valid	Deposit	Valid	Skip	Skip	Skip	Skip	Skip	IncreaseBalance	IncreaseBalance	Send	NotPrint	Send		
Valid	Deposit	Valid	Skip	Skip	Skip	Skip	Skip	IncreaseBalance	IncreaseBalance	Send	Print	Send		
Valid	Deposit	Skip	Skip	Skip	Skip	Skip	Skip	IncreaseBalance	IncreaseBalance	Send	NotPrint	Send		
Valid	Deposit	Skip	Skip	Skip	Skip	Skip	Skip	IncreaseBalance	IncreaseBalance	Send	NotPrint	Send		
Valid	Deposit	Valid	Skip	Skip	Skip	Skip	Skip	IncreaseBalance	IncreaseBalance	Send	Print	Send		
Valid	Withdraw	Valid	Skip	Skip	Skip	Skip	Skip	DecreaseBalance	DecreaseBalance	Send	NotPrint	Send		
Valid	Withdraw	Valid	Skip	Skip	Skip	Skip	Skip	DecreaseBalance	DecreaseBalance	Send	Print	Send		
Valid	Transfer	Valid	Select	Valid	Skip	Skip	ValidInput	Agree	Skip	SendMoney	DecreaseBalance	Send	NotPrint	Send
Valid	Transfer	Valid	Select	Valid	Skip	Skip	ValidInput	Disagree	Skip	SendMoney	DecreaseBalance	Send	NotPrint	Send
Valid	Transfer	Valid	Select	Valid	Skip	Skip	ValidInput	Agree	Skip	SendMoney	DecreaseBalance	Send	Print	Send
Valid	Transfer	Valid	Select	Valid	Skip	Skip	ValidInput	Disagree	Skip	SendMoney	DecreaseBalance	Send	Print	Send
Valid	IssueTC	Valid	Skip	Skip	Valid	Skip	Skip	Skip	Send	DecreaseBalance	IncreaseBalance	Send	Print	Send
Valid	IssueTC	Valid	Skip	Skip	Valid	Skip	Skip	Skip	Send	DecreaseBalance	IncreaseBalance	Send	NotPrint	Send

## Test Cases & Result

CPT와 마찬가지로 프로그램 시작 부분에 에러.

→ 수정 전 테스트 결과: **All Fail**



## Test Cases & Result (코드 수정 후)

No.	TestCase	Result (2nd)	Description
1	100.200.300.1401.1500	Passed	
2	100.200.300.1400.1500	Failed	영수증 용지가 부족한 경우 알람이 오지 않음.
3	100.200.301	Passed	
4	100.201.802.1100.1200.1300.1400.1500	Failed	50000원을 눌렀을 때 화면에 5000원이 표시됨.
5	100.201.300.802.1100.1200.1300.1401.1500	Failed	50000원을 눌렀을 때 화면에 5000원이 표시됨.
6	100.201.300.803.1100.1200.1300.1400.1500	Failed	50000원을 눌렀을 때 화면에 5000원이 표시됨.
7	100.201.801.1100.1200.1300.1401.1500	Failed	현금이 부족한 경우 알람이 오지 않음.
8	100.201.803.1100.1200.1300.1401.1500	Failed	50000원을 눌렀을 때 화면에 5000원이 표시됨.
9	100.201.300.801.1100.1200.1300.1400.1500	Failed	현금이나 영수증 용지가 부족한 경우 알람이 오지 않음.
10	100.201.301	Passed	
11	100.202.300.700.800.1100.1201.1300.1401.1500	Failed	한 자리 또는 두 자리의 돈을 정상적으로 출금할 수 없음.
12	100.202.300.700.800.1100.1201.1300.1400.1500	Failed	한 자리 또는 두 자리의 돈을 정상적으로 출금할 수 없음.



## Test Cases & Result (코드 수정 후)

13	100.202.300.700.804	Failed	돈을 0원 또는 0달러 출금시 수수료만 요구
14	100.202.301	Failed	돈을 0원 또는 0달러 출금시 수수료만 요구
15	100.203.300.400.500.800.900.1102.1201.1300.1401.1500	Failed	대부분의 계좌가 조회되지 않음.
16	100.203.300.400.500.800.901.1102.1201.1300.1401.1500	Failed	대부분의 계좌가 조회되지 않음.
17	100.203.300.400.500.800.900.1102.1201.1300.1400.1500	Failed	대부분의 계좌가 조회되지 않음.
18	100.203.300.400.500.800.901.1102.1201.1300.1400.1500	Failed	대부분의 계좌가 조회되지 않음.
19	100.203.300.400.501	Passed	
20	100.203.300.400.500.804	Failed	대부분의 계좌가 조회되지 않음.
21	100.203.301	Passed	
22	100.204.300.600.1000.1100.1200.1300.1400.1500	Failed	교통카드 잔량이나 현금이 부족한 경우 알람이 오지 않음.
23	100.204.300.600.1000.1100.1200.1300.1401.1500	Failed	교통카드 잔량이나 현금이 부족한 경우 알람이 오지 않음.
24	100.204.301.601	Failed	유효기간이 01부터 시작해야 하나 00인 경우도 발급됨.
25	100.204.301	Passed	

→ 6 / 25 (24%) Pass

- ✓ 1<sup>st</sup> Pairwise Testing => 32% Passed
- ✓ 2<sup>nd</sup> Pairwise Testing => 24% Passed

=> 첫번째 PT보다 낮아진 Passed Percentage

## Pairwise Testing Summary

- 파일 입출력 과정에서의 경로 / 자료구조 문제 심각.
- alarm 기능 작동하지 않음.
- 예외처리 미흡.
- 약간의 UI 오류.
- 수수료 등 추가된 일부 기능에서 에러 발생

Test	Test Num	기존 P/F	현재 P/F	비고
아이템 받아오기	1-1	Fail (Pass)	Fail (Pass)	(변화 없음)
	1-2	Fail (Pass)	Fail (Pass)	(변화 없음)
	1-3	Fail (Fail)	Fail (Pass)	수정 확인.
	1-4	Fail (Fail)	Fail (Pass)	수정 확인.
	1-5	Fail (Fail)	Fail (Pass)	수정 확인.
입금	2-1	Fail (Fail)	Fail (Pass)	수정 확인. (GitHub issue tracker 내용 확인 요함)
	2-2	Fail (Pass)	Fail (Pass)	(변화 없음)

출금	3-1	Fail (Fail)	Fail (Pass)	수정 확인.
	3-2	Fail (Pass)	Fail (Pass)	(변화 없음)
	3-3	Fail (Fail)	Fail (Fail)	UI에 정상적으로 출력되지 않는다.
	3-4	Fail (Fail)	Fail (Fail)	999만원 까지 밖에 출금을 진행할 수 없다.
계좌이 체	4-1	Fail (Pass)	Fail (Pass)	(변화 없음)
	4-2	Fail (Fail)	Fail (Pass)	수정 확인.
	4-3	Fail (Fail)	Fail (Fail)	999만원 까지 밖에 이체를 진행할 수 없다.
	4-4	Fail (Pass)	Fail (Pass)	(변화 없음)
	4-5	Fail (Pass)	Fail (Pass)	(변화 없음)

알림		Fail (Fail)	Fail (Fail)	메일 기능 비정상. (신규 오류에 추가)
	5-2	Fail (Fail)	Fail (Fail)	메일 기능 비정상. (신규 오류에 추가)
	5-3	Fail (Fail)	Fail (Fail)	메일 기능 비정상. (신규 오류에 추가)
	5-4	Fail (Fail)	Fail (Fail)	메일 기능 비정상. (신규 오류에 추가)
교통카드	6-1	Fail (Fail)	Fail (Fail)	3달러가 출금됨.
	6-2	Fail (Fail)	Fail (Pass)	수정 확인.

0 / 22 = 0% Passed (15 / 22 = 68.18% Passed)

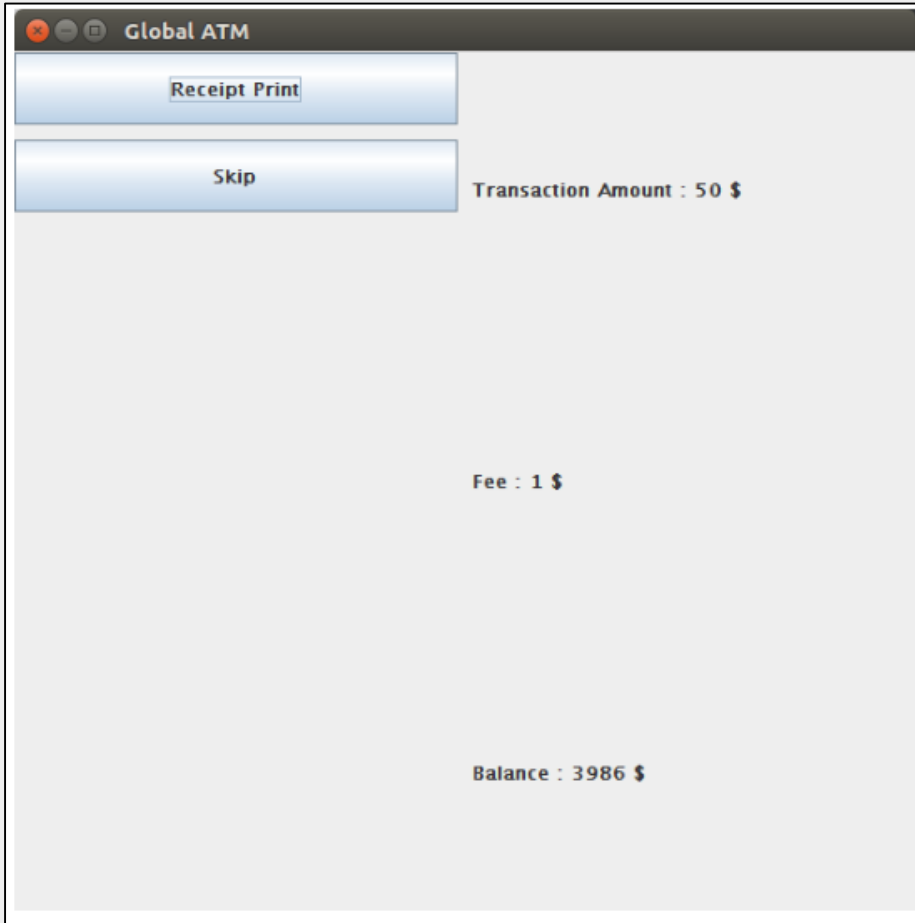
## Why?

- ✓ FileNotFoundException 은 2주 연속으로 문제가 발생하는 것을 확인함.
- ✓ 따라서 Pass percentage는 0% 로 책정이 가능함.
- ✓ 괄호 안은 FileNotFoundException을 제외한 결과임.

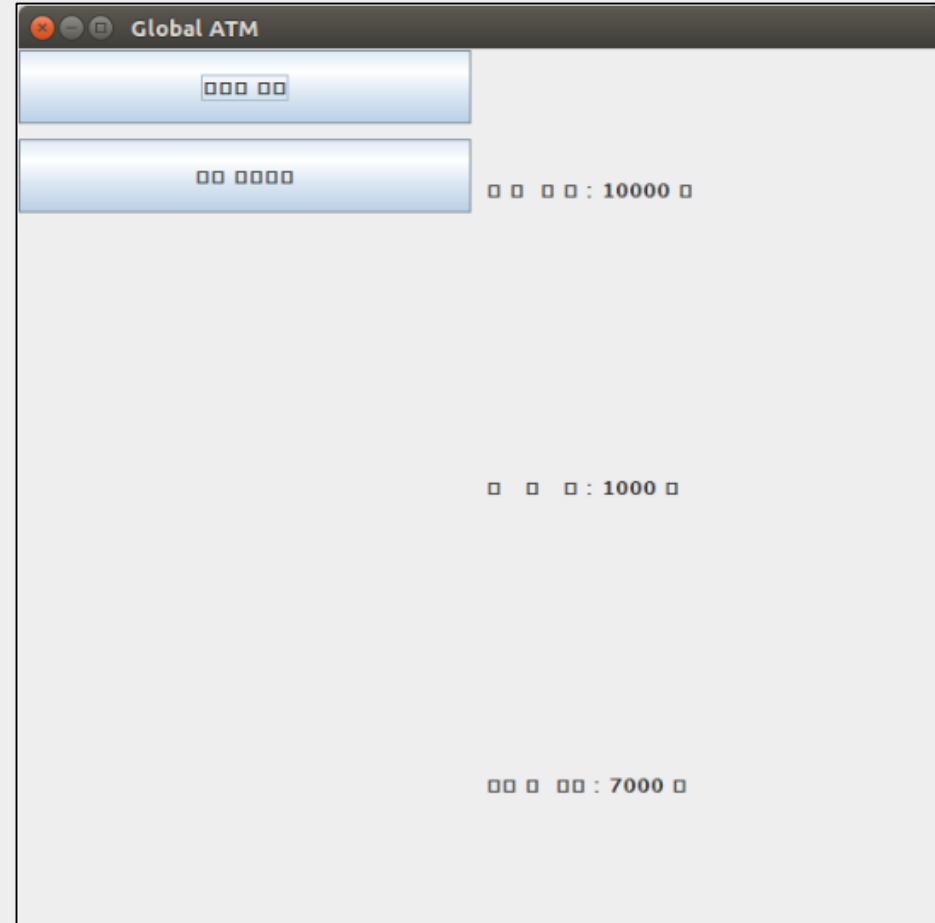
## 1. Deposit 진행 시 mail 발송 관련 Exception이 출력됨.

```
javax.mail.AuthenticationFailedException: 535-5.7.8 Username and Password not accepted.  
Learn more at  
535 5.7.8 https://support.google.com/mail/?p=BadCredentials y20-v6sm25785168pge.46 -  
gsmtip  
  
    at  
com.sun.mail.smtp.SMTPTransport$Authenticator.authenticate(SMTPTransport.java:823)  
    at com.sun.mail.smtp.SMTPTransport.authenticate(SMTPTransport.java:756)  
    at com.sun.mail.smtp.SMTPTransport.protocolConnect(SMTPTransport.java:673)  
    at javax.mail.Service.connect(Service.java:317)  
    at javax.mail.Service.connect(Service.java:176)  
    at javax.mail.Service.connect(Service.java:125)  
    at javax.mail.Transport.send0(Transport.java:194)  
    at javax.mail.Transport.send(Transport.java:124)  
    at ATM.ATM.checkResource(ATM.java:631)  
    at GUI.GUIController.printReceipt(GUIController.java:923)  
    at GUI.GUIController$22.actionPerformed(GUIController.java:1143)  
    at  
java.desktop/javafx.swing.AbstractButton.fireActionPerformed(AbstractButton.java:1967)  
    at  
java.desktop/javafx.swing.AbstractButton$Handler.actionPerformed(AbstractButton.java:2308)  
    at  
java.desktop/javafx.swing.DefaultButtonModel.fireActionPerformed(DefaultButtonModel.java:4  
05)  
    at  
java.desktop/javafx.swing.DefaultButtonModel.setPressed(DefaultButtonModel.java:262)  
    at  
java.desktop/javafx.swing.plaf.basic.BasicButtonListener.mouseReleased(BasicButtonListener  
.java:270)  
    at java.desktop/java.awt.Component.processMouseEvent(Component.java:6589)  
    at
```

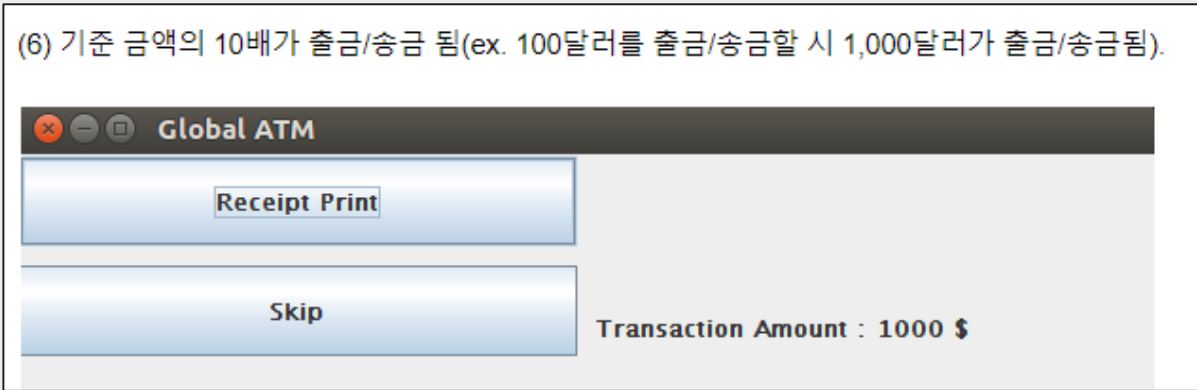




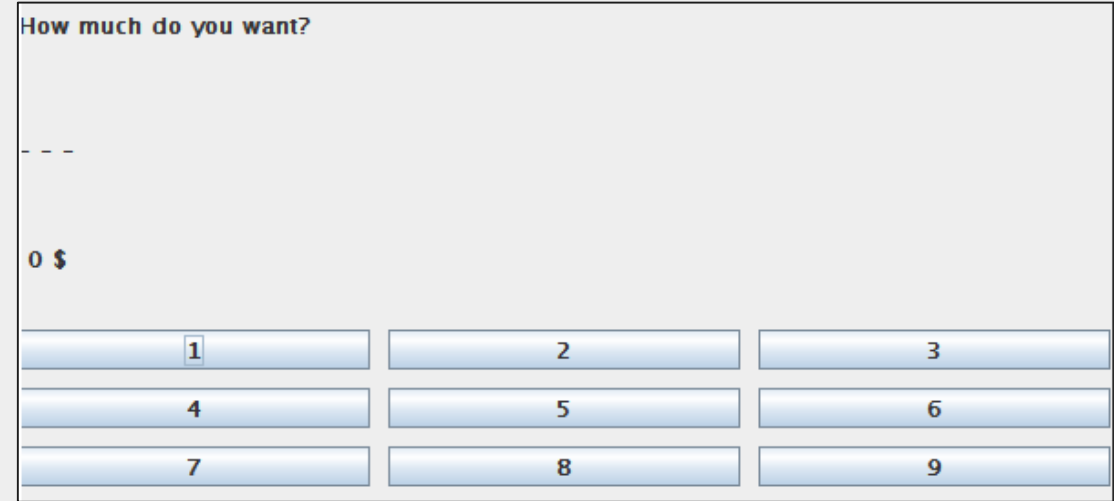
2. 외국 은행에서 withDraw 진행 시 무조건 달러로 출금된다.



2. 국내 은행에서 withDraw 진행 시 무조건 원화로 출금된다.



▲ 4-2. 외화를 출금 시 10배의 금액이 출금된다.



▲ 4-3. 세 자리의 숫자만 출금이 가능하다.

Transaction Amount : 3 \$

▲ 6-1. 교통카드의 발급비용은 0.3\$라고 명시되어 있지만 3\$가 출금된다.

# 04. Static Analysis

101 Open ✓ 16 Closed

Author ▾ Labels ▾ Projects ▾ Milestones ▾ Assignee ▾ Sort ▾

- #21 by kusv2018t1 was closed an hour ago System Test  
[BruteForceTest] ATM기기가 교통카드 여분을 보유하고 있지 않을 때 교통카드를 생성한다. Bug Confirm Fixed
- #19 by kusv2018t1 was closed an hour ago System Test  
[BruteForceTest] ATM기기가 1000원을 보유하고 있지 않을 때 23000원의 출금을 시도한다. Bug Confirm Fixed
- #18 by kusv2018t1 was closed an hour ago System Test  
[BruteForceTest] ATM기기가 10\$를 최대로 보유하고 있을 때 10\$의 입금을 시도한다. Bug Confirm Fixed
- #17 by kusv2018t1 was closed an hour ago System Test  
[BruteForceTest] ATM기기가 10000원을 최대로 보유하고 있을 때 10000원의 입금을 시도한다. Bug Confirm Fixed
- #16 by kusv2018t1 was closed an hour ago System Test  
[BruteForceTest] 명세표 용지가 남아있지 않을 때 명세표를 출력한다. Bug Confirm Fixed
- #14 by kusv2018t1 was closed an hour ago System Test  
[BruteForceTest] 50달러를 이체한다. Bug Confirm Fixed
- #12 by kusv2018t1 was closed an hour ago System Test  
[BruteForceTest] 50달러를 출금한다. Bug Confirm Fixed
- #11 by kusv2018t1 was closed an hour ago System Test  
[BruteForceTest] ATM기기가 보유하고 있는 50000원권이 없을 때 50000원을 출금한다. Bug Confirm Fixed

Github issue Tracker를  
이용한 이슈관리

<b>Bug</b>	[SV전용]프로그램의 버그에 대한 issue의 경우 적용하여 주십시오.
<b>Confirm</b>	[SV전용]검수가 완료되어 Closed 예정인 issue에 대해 적용하여 주시기 바랍니다.
<b>Duplicate</b>	중복 issue일 경우 이 label을 적용하여 주세요. 몇 번 issue와 중복인지 꼭 comment를 달아주시기 바랍니다.
<b>Fixed</b>	해결이 완료되었을 때 이 label을 적용하여 주세요. branch name을 달아주시기 바랍니다.
<b>WontFix</b>	해결하지 않을/못할 issue의 경우 이 label을 적용하여 주세요. 꼭 이유를 달아주시기 바랍니다.

◀ label을 활용한 소통

▼ milestone 활용

🚩 1 Open ✓ 0 Closed Sort ▾

---

**System Test**  12% complete 101 open 14 closed

⚠ Past due by 2 days ⌚ Last updated about 1 hour ago

### Build History

find

Build ID	Time	Status
#92	2018. 5. 30 오전 8:32	Success
#91	2018. 5. 30 오전 2:04	Failed
#90	2018. 5. 29 오후 4:00	Failed
#89	2018. 5. 29 오전 11:07	Failed
#88	2018. 5. 29 오전 8:57	Success
#87	2018. 5. 29 오전 8:13	Success
#86	2018. 5. 28 오전 10:26	Success
#85	2018. 5. 28 오전 8:17	Success

### kusv2018t1

Issues Measures Code Activity Administration

Quality Gate **Failed**

- Security Rating on New Code: **B** (is worse than A)
- Reliability Rating on New Code: **E** (is worse than A)
- Coverage on New Code: **0.0%** (is less than 80.0%)

Bugs & Vulnerabilities

Category	Count	Quality
Bugs	6	<b>E</b>
Vulnerabilities	13	<b>B</b>
New Bugs	6	<b>E</b>

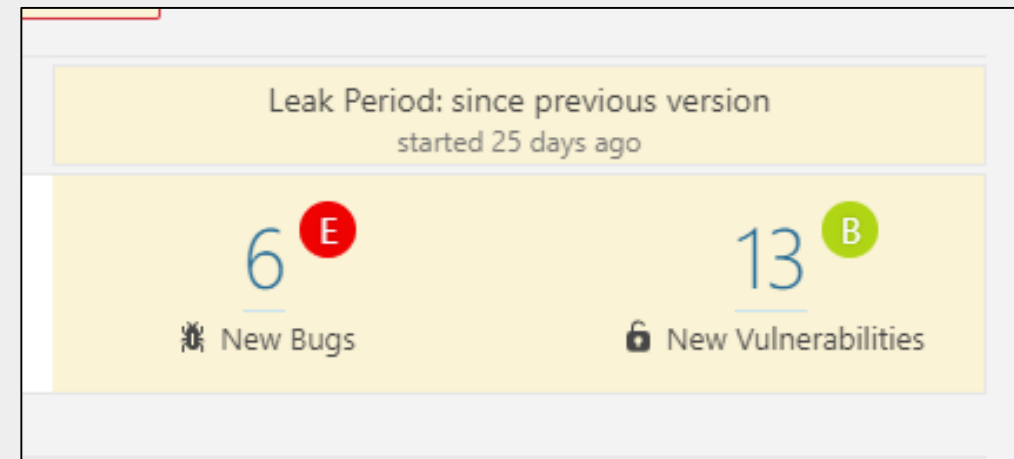
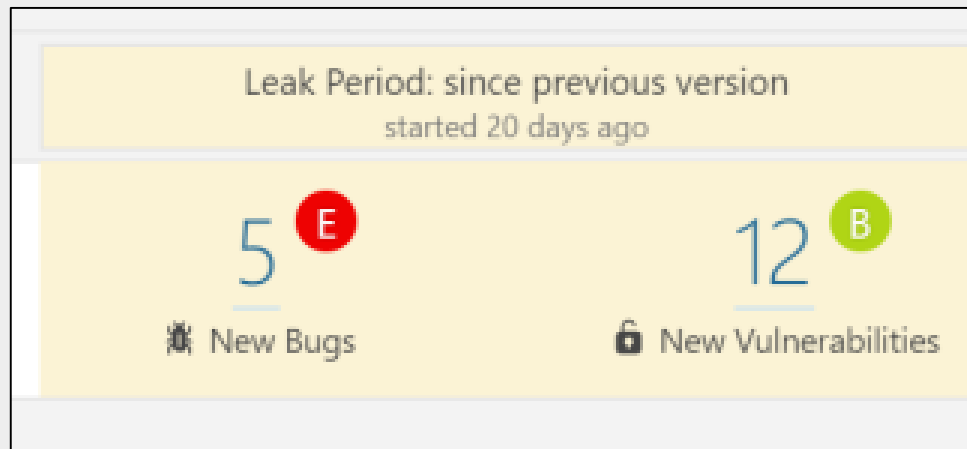
Leak Period: since started 25

### Type

Bug	6
Vulnerability	13
Code Smell	452

92번째 Build를 기준으로 진행

이전 Build와의 비교




기존의 Bug, Vulnerabilities 전혀 해결되지 않음  
Bug, Vulnerability의 개수 늘어남



```
transactionAmount = service;
switch (service) {
    // 계좌조회
    case 1:
        break;
    // 입금
    case 2:
        break;
    // 출금
    case 3:
        break;
    // 송금
    case 4:
        break;
    // 교통카드 발급
    case 5:
        break;
}
return;
```

Remove this conditional structure or edit its code blocks so that they're not all the same. [...](#) 8 days ago ▾ L139 

 Bug ▾  Major ▾  Open ▾ Not assigned ▾ 15min effort [Comment](#)  No tags ▾









```
//mention delete  
br.readLine();
```

Use or store the value returned from "readLine" instead of throwing it away. ...

4 days ago ▾ L64 

 Bug ▾  Major ▾  Open ▾ Not assigned ▾ 5min effort [Comment](#)

 No tags ▾

printStackTrace() 사용

```
e.printStackTrace();
```

Use a logger to log this exception. ...

Code scroll에서도 발견되는 Vulnerability 문제

## 패키지명, 변수명 규칙






Rename this package name to match the regular expression `^[a-z]+(\.[a-z][a-z0-9]*)*$`. ...

 Code Smell  Minor  Open  Not assigned  10min effort [Comment](#)

```
import BankSystem.Bank;  
import Item.TrafficCard;
```

```
}  
//스브? : 환화도 아니고 달라도 아니고?
```

This block of commented-out lines of code should be removed. ...

 Code Smell  Major  Closed (Fixed)  Not assigned  5min effort [Comment](#)

```
private int max_cash = 1000;
```

Rename this field "max\_cash" to match the regular expression `^[a-z][a-zA-Z0-9]*$`. ...

 Code Smell  Minor  Open  Not assigned  2min effort [Comment](#)

Type	Method
Name	Agreement
Purpose	교통 카드 발급 약관을 보여주고, 서명을 받는다.

Type	Method
Name	destAccount
Purpose	송금할 대상을 설정한다.

Type	Method
Name	get_name
Purpose	해당 계좌의 이름을 가져온다.

Type	Method
Name	getATMadminID
Purpose	관리자 ID 를 return 해준다.

Methods	<u>Methods should be verbs</u> , in mixed case with the first letter lowercase, with the first letter of each internal word capitalized.	run(); runFast(); getBackground();
---------	--	--

출처: 오라클 홈페이지

(<http://www.oracle.com/technetwork/java/codeconventions-135099.html>)

Type	Method
Name	Agreement
Purpose	교통 카드 발급 약관을 보여주고, 서명을 받는다.

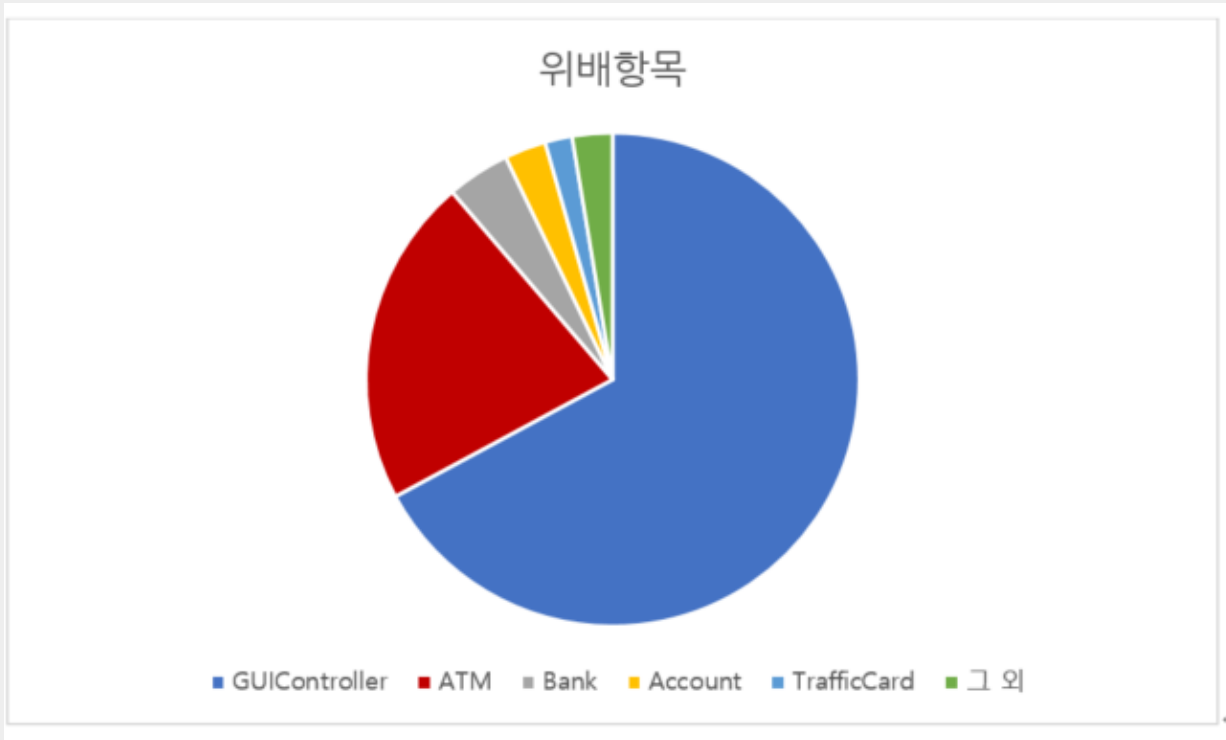
Type	Method
Name	get_name
Purpose	해당 계좌의 이름을 가져온다.

Type	Method
Name	destAccount
Purpose	송금할 대상을 설정한다.

Type	Method
Name	getATMadminID
Purpose	관리자 ID 를 return 해준다.

Methods	Methods should be verbs, in mixed case with <u>the first letter lowercase, with the first letter of each internal word capitalized.</u>	run(); runFast(); getBackground();
---------	---	--

출처: 오라클 홈페이지  
 (http://www.oracle.com/technetwork/java/codeconventions-135099.html)



소스	위배 수	위배 규칙 수	*RCR	**RV D
GUIController.java	715	19	80.81%	0.36
ATM.java	230	19	80.81%	0.47
Bank.java	44	12	87.88%	0.27
Account.java	29	7	92.93%	0.46
TrafficCard.java	19	11	88.89%	0.37
Card.java	15	7	92.93%	0.34
Book.java	13	6	93.94%	0.32

GUIController.java 에서 가장 많은 위배 항목 발견

규칙	위배 수	규칙 설명
Sun_24	581	숫자 상수 사용 금지
Sun_08	105	블록의 시작 부분에서만 선언 검사
Sun_10	72	지역 변수 선언 시 초기화 검사
JAVA_72	63	메서드의 설명 주석이 있어야 함
JAVA_64	42	반복문 안에서 String concatenation assignment 연산자의 사용 금지
Sun_06	26	소스 라인 길이 검사
Sun_17	20	메서드 내부의 지역 변수 선언부와 실행 문장 사이를 빈 줄로 구분

주로 소스코드의 변수 이름과 관련된 부분에서 위배가 발견됨

심각도	위배수	포함된 규칙 수	위배 규칙 수	*SCR	설명
매우 높음	13	20	1	95%	프로그램에 매우 심각한 영향을 미치므로 오류 수정이 필요함
높음	135	46	6	86.96%	프로그램에 심각한 영향을 줄 수 있으므로 강력히 권장됨
낮음	104	10	4	60%	프로그램에 영향을 줄 수 있으므로 오류 수정이 권장됨
매우 낮음	813	23	14	39.13%	프로그램에 큰 영향은 없지만 오류 수정이 권장됨



## 메서드 (printStackTrace)가 사용됨

```
90 :           getStr = br.readLine();
91 :           getMailID = getStr;
92 :
93 :           getStr = br.readLine();
94 :           rate = Integer.parseInt(getStr);
95 :       } catch (FileNotFoundException e) {
96 :           e.printStackTrace();
```

[JAVA\_21] catch 절에서 exception(java.io.FileNotFoundException)이 그대로 출력됨.

[ERR01-J] catch 절에서 (java.io.FileNotFoundException)의 메서드를 사용함

[JAVA\_44] 메서드 (printStackTrace)가 사용됨.

[ERR00-J] catch 절에 printStackTrace 메서드만 존재

반복문 안에서 String concatenation 연산자를 사용함

```
801 : //print pressed amount
802 : for(int j = 0; j < pw_i; j++){
803 :     ptn_s += "X" + " ";
```

[JAVA\_64] 반복문 안에서 String concatenation 연산자를 사용함.

Figure 3 GUIController.java line 803

Identifier(pth\_s)가 이미 필드로 선언되어 있어 hide가 발생함

```
645 : String ptn_s = "";
```

[Sun\_09] identifier(pth\_s)가 이미 필드로 선언되어 있어 hide가 발생함

[JAVA\_70] identifier(pth\_s)가 이미 필드로 선언되어 있어 hide가 발생함

Figure 3 GUIController.java line 646

# 05. Summary

Category-Partition Test →  $6 / 25 = 24\%$  Pass

Pairwise Test →  $15 / 64 = 25\%$  Pass

Brute-force Test →  $15 / 22 = 62\%$  Pass

- ✓ Text File의 경로가 잘못되어 실행이 되지 않는 Error 발생
- ✓ Text File을 Parsing하는 과정에서 Array Index Out of Bounds Error 발생
- ✓ 관리자에게 Alarm을 보내야 하는 모든 경우 실패
  1. ATM에 돈이 부족한 경우
  2. ATM에 돈이 너무 많은 경우
  3. 영수증이 부족한 경우
  4. 교통카드가 부족한 경우
- ✓ User Interface 오류
  1. 입금시 금액 출력 오류 (50000원 -> 5000원)
  2. 출금시 금액 입력 화면에 확인 버튼이 없음

- ✓ 메일 전송 기능이 제대로 동작하지 않는다.
- ✓ 여러 경우에 대해 정확한 예외처리가 되어있지 않다.
- ✓ 위와 같은 상황 등 Error가 많으므로 수정 필요