



# C++, The Convenient ATM

---

T2

201411262 김도현

201411271 박상우

201411312 장하나

201411316 정진호



# CONTENTS

1. System Test Result
  2. Specification Revision
  3. System Test 대응
-

# 1. System Test Result

- Stage 2030 Analysis 에 예외 상황이 생략되어 있음.
  - Stage 2030 Analysis 에 불필요한 Cross Reference
  - Stage 2050 Implementation 에 용어 불일치
  - Stage 2060 Testing 에 Fail이 Pass로 잘못 기록
  - Category-partition Testing **78%** PASS
  - Pairwise Testing **78%** PASS
  - Brute Force Testing **65%** PASS
-

## 2. Specification Revision

문제	Typical Course of Events	<p>(A) : Actor, (S) : System</p> <ol style="list-style-type: none"> <li>(A) 계좌 정보를 입력한다.</li> <li>(S) 계좌 정보의 유효성을 확인한다. (E1)</li> <li>(A) 계좌 비밀번호를 입력한다.</li> <li>(S) 비밀번호가 일치하는지 확인한다.</li> <li>(S) 계좌의 잔액을 출력한다.</li> </ol>
	Alt. Course of Events	N/A
	Exceptional Course of Events	<p>E1. 계좌 정보가 유효하지 않다면 진행되지 않는다.  E2. 잘못된 비밀번호를 입력한다며 진행되지 않는다.</p>
대응	<p>Typical Course of Events 중 "비밀번호가 일치하는지 확인한다" 부분에 E2 예외 상황을 추가.</p>	

## 2. Specification Revision

문제	Use Case	9. Calculate Fee
	Actors	User
	Purpose	거래의 종류에 따라 수수료를 계산한다.
	Overview	거래 종류와 은행에 따른 수수료를 계산한다. 수수료만큼의 금액을 해당 계좌에서 차감한다.
	Type	Primary
	Cross References	Functions : R5, R1.1, R1.2, R2.1, R2.2, R3.1, R3.2, R4.1, R4.2
대응	해당 Cross Reference 삭제	

## 2. Specification Revision

문제	Remove Admin / Toggle ATM State / Query ATM Balance 가 Actor-based Use Case인데 Typical Course of Events에 Actor의 행동이 없음.
대응	해당 Use Case는 Actor의 행동을 따로 기술할 필요 없다 판단.

## 2. Specification Revision

문제	<p>이체라고 되어있으나 UI에는 송금이라고 되어있음.</p> <table border="1" data-bbox="355 562 1717 701"><tr><td data-bbox="355 562 724 629">Name</td><td data-bbox="724 562 1717 629">1. press "이체"</td></tr><tr><td data-bbox="355 629 724 701">Responsibilities</td><td data-bbox="724 629 1717 701">메뉴 선택 화면의 "이체" 버튼을 누른다.</td></tr></table>	Name	1. press "이체"	Responsibilities	메뉴 선택 화면의 "이체" 버튼을 누른다.
Name	1. press "이체"				
Responsibilities	메뉴 선택 화면의 "이체" 버튼을 누른다.				
대응	이체로 수정함				

## 2. Specification Revision

문제	Test 결과 Fail인 경우가 Pass로 되어있음.
대응	환율을 정확히 받아오는 기능 : 정상작동함이 확인됨 다양한 상황을 테스트해보지 못함.

### 3. System Test 대응

문제	<p>입금 과정에서 계좌번호 및 비밀번호 입력 후, 입금 단계 도중 atm의 지폐 보유량을 변경하고 다시 입금창으로 돌아와 지폐 장수를 입력하면 입금이 되지 않는다.</p> <p>이 상태에서 입금창의 취소 버튼을 누르면 메인화면이 관리자 기능 선택 화면으로 변한다.</p>
원인	<p>User와 Admin의 기능을 동시에 실행할 수 있기 때문에 발생한다.</p>

### 3. System Test 문제 원인

```
switch (system.getFunction()) {  
    case Deposit:  
        try {  
            system.enterBill(values);  
        } catch (DataStoreError | InvalidBillException ex) {  
            ex.printStackTrace();  
            parentFrame.setContentPane(new SelectFunction(parentFrame));  
            parentFrame.pack();  
            parentFrame.invalidate();  
            parentFrame.validate();  
            return;  
        }  
}
```

```
switch (system.getFunction()) {  
    case Withdraw:  
        if (insertNumber > 100) {  
            JOptionPane.showMessageDialog(parentFrame, message);  
            return;  
        } else if (insertNumber == 0) {  
            JOptionPane.showMessageDialog(parentFrame, message);  
            return;  
        }  
  
        try {  
            system.enterBillAmountToWithdraw(cashAmount: -insertNumber);  
        } catch (DataStoreError | NegativeBalanceError ex) {  
            JOptionPane.showMessageDialog(parentFrame, ex.getMessage());  
            parentFrame.setContentPane(new SelectFunction(parentFrame));  
            parentFrame.pack();  
            parentFrame.invalidate();  
            parentFrame.validate();  
            return;  
        }  
}
```

### 3. System Test 대응

대응

User 또는 Admin 중 하나가 기능을 실행하고 있는 상태에서는 다른 한쪽의 작동을 막는다.

### 3. System Test 대응

문제	atm 지폐 보유량 한도를 초과하는 입금을 시도할 경우 거래가 취소된 후 atm이 잠기며, 시도한 입금만큼 atm 지폐 보유량이 증가한다.
원인	ATM 잔고에 입금된 지폐를 더한 후, User 쪽에 예외를 발생시켜서 문제가 발생함

### 3. System Test 대응

대응

사전에 지폐량을 더하여, 값을 확인한 후, 예외가 발생하지 않으면, ATM 잔고에서 값을 받도록 수정하였다

### 3. System Test 대응

문제	지폐 입력 단계에서 유효하지 않은 값을 입력한 뒤(ex. 문자열) 입력을 누르면 경고창이 실행되지 않는다
원인	UI에서 문자열 입력을 허용한다.

### 3. System Test 대응

```
public class JNumberTextField extends JTextField {
```

```
    thousandField : JNumberTextField (com.swac  
    thousandLabel : JLabel  
    fiveThousandField : JNumberTextField (com.  
    fiveThousandLabel : JLabel
```

₩ 1000

```
    thousandField:JNumberTextField
```

```
@Override  
public void processKeyEvent(KeyEvent ev) {  
    if (Character.isDigit(ev.getKeyChar()) || ev.getKeyCode() == KeyEvent.VK_BACK_SPACE) {  
        super.processKeyEvent(ev);  
    }  
    ev.consume();  
}
```

기존 컴포넌트 확장



제약 추가

### 3. System Test 대응

문제	메인화면/관리창에서 각각 언어 변경을 했을 때 다른 창에 즉시 적용되지 않는다.
원인	UI에서 System의 속성을 관찰할 수 없어서 두 창을 한꺼번에 바꾸는 것이 불가능하여 발생하였다.

### 3. System Test 대응



```
parentFrame.setContentPane(new SelectFunction(parentFrame, system).getPanel());  
parentFrame.pack();  
parentFrame.invalidate();  
parentFrame.validate();
```

Locale 변경 후 Frame Refresh 필요

### 3. System Test 대응

```
private Main() {
    super();
    userFrame = new JFrame( title: "C++, The Convenient ATM");
    adminFrame = new JFrame( title: "Admin Interface for C++ ATM");
}

public static Main getInstance() {
    if ( application == null ) {
        application = new Main();
    }

    return application;
}
```

Main Class를 Singleton으로

### 3. System Test 대응

```
public void initializeScreen() {  
    this.userFrame.setVisible(false);  
    this.userFrame.pack();  
    this.userFrame.setContentPane(new SelectFunction(application.userFrame, system).getContentPane());  
    this.userFrame.invalidate();  
    this.userFrame.validate();  
    this.userFrame.setVisible(true);  
  
    this.adminFrame.setVisible(false);  
    this.adminFrame.pack();  
    this.adminFrame.setContentPane(new AuthorizeAdmin(application.adminFrame, system).getContentPane());  
    this.adminFrame.invalidate();  
    this.adminFrame.validate();  
    this.adminFrame.setVisible(true);  
}
```

```
koreanButton.addMouseListener((MouseAdapter) m  
    .. system.changeLocale(Locale.ko_KR);  
    .. application.initializeScreen();  
});
```

Locale 변경 시 모든 Frame 초기화

### 3. System Test 대응

문제	<p>비밀번호 또는 연락처를 입력하지 않아도 admin 계정이 생성된다. 형식에 맞는 입력에 대한 정확한 정의가 필요하다. 문서상 유효하지 않은 값은 입력 시 초기화면으로 돌아간다고 표기되어 있다.</p>
원인	<p>enterAdminInfo에서 입력된 관리자 정보 (비밀번호, 연락처)에 대한 예외처리가 생략되어 있어서 발생하였다.</p>

### 3. System Test 대응

```
public Admin(String id, String password, String contact) throws InvalidAdminException {
    if (!this.checkPassword(password) || !this.checkContact(contact)) {
        throw new InvalidAdminException();
    }

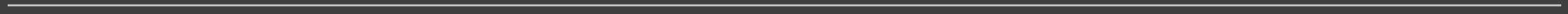
    this.id = id;
    this.password = password;
    this.contact = contact;
}
```

```
    return Pattern.matches(regex: "\\d{4}", password); }
    return Pattern.matches(regex: "(0\\d{8,10})", contact); }
```

Admin 속성에 대한 정규표현식  
추가



# 2차 시연





# Q & A

