

# Software Modeling & Analysis

## Global ATM System

### Project Team

1 Team

Date

2018-03-31

---

### Team Information

201311287 엄현식

201311318 최정현

201611293 전다운

## 목차

Activity 1001. Define Draft Plan .....	4
1. Motivation.....	4
2. Project Objectives .....	4
3. Functional Requirements.....	4
4. Non-Functional Requirements.....	4
5. Resource Estimation.....	5
6. Other Information .....	5
Activity 1002. Create Preliminary Investigation Report .....	5
1. Alternative Solutions .....	5
2. Project Justification.....	5
3. Risk Management .....	6
4. Risk Reduction plan.....	6
5. Market Analysis.....	7
6. Other Managerial Issues .....	7
Activity 1003. Define Requirements .....	7
1. Functional Requirements.....	7
2. Performance Requirements .....	8
3. Operating Environments.....	8
4. Development Environments .....	8
Activity 1004. Record Terms in Glossary .....	8
Activity 1006. Define Draft System Architecture.....	8
Activity 1007. Define Business Use Case .....	9
1 Determine System boundary .....	9
2 Identify and Describe Actors.....	9
3 Identify Use-Case .....	10

4	Allocate system functions into Related Use-Cases.....	10
5	Categorize Use-Cases.....	11
6	Identify the relationships between Use-Case.....	11
7	Draw a Use-Case diagram.....	13
8	Describe Use-Case.....	13
9	Rank Use-Case.....	15
	Activity 1008. Define Business Concept Models.....	16
	Activity 1009. Test Case.....	16
	Activity 1010. Refine Plan.....	17
1	Project Scope.....	17
2	Project Objectives.....	17
3	Functional Requirements.....	17
4	Performance Requirements.....	18
5	Operating Environment.....	18
7	Scheduling.....	18
8.	Configuration Management.....	19
9.	Quality Assurance Plan.....	19

## Activity 1001. Define Draft Plan

### 1. Motivation

작년 7월 카카오 은행의 등장으로 기존 금융업계가 크게 흔들리고 있다. 금리, 수수료 면제는 물론 스마트폰만 있다면 쉽고 간편한 송금, 조회, 입금, 출금이 실현 되었기 때문이다. 이러한 핀테크(fintech)\*의 발전으로 기존 오프라인 은행의 필요성마저 사라지고 있다. 이 흐름속에서 ATM은 어떻게 변화해야 할까?

이번 설 연휴의 해외여행객은 4만 4700명으로 작년보다 15.5%나 증가 했다고 한다. 또한 평창올림픽의 여파로 국내로 여행오는 외국여행객도 증가하고 있다. 핀테크의 위협에서 안전하고 여행객들의 수요를 충족할 수 있는 '환전'을 ATM과 접목, 인천공항을 방문하는 여행객들의 환전을 쉽고 빠르게 돕는 Ex ATM(Exchange ATM)을 개발하게 되었다.

\*핀테크(fintech) : '금융(finance)'과 '기술(technology)'이 결합한 서비스를 뜻함

### 2. Project Objectives

- 안전하고 빠른 현금거래(입/출금)를 수행한다.
- 인천공항을 방문하는 여행객(한국인,외국인)이 쉽고 빠르게 환전하도록 돕는다.

### 3. Functional Requirements

- Read Item
- Confirm
- Check
- Deposit
- Withdraw
- Transfer
- Update
- Receipt Print
- Status Alarm
- Cash Management

### 4. Non-Functional Requirements

- 외국인, 내국인이 쉽게 이해할 수 있는 직관적인 인터페이스여야 한다.
- 적절한 본인인증을 통해, 타의 혹은 악의적인 접근을 막을 수 있어야 한다.
- ATM은 항상 적정량의 현금을 소지하고 있어야 한다.

## 5. Resource Estimation

- 1) Human Resource : 3 명
- 2) Project Duration : 3 개월
- 3) Human Efforts : 12
- 4) Cost : 3,000,000 원

## 6. Other Information

- 현재 환율정보를 고려하여 환전을 돕는다.
- 명세표를 문자로 확인할 수 있다.
- 현금 인출기의 오류, 현금부족, 현금포화 상태의 경우 관리자에게 문자 알림을 한다.

## Activity 1002. Create Preliminary Investigation Report

### 1. Alternative Solutions

- 1) ATM 제작 업체에게 제안한다.
- 2) 개발 업체에게 제작을 의뢰한다.
- 3) 직접 개발한다.
- 4) 직접 개발한다' 를 선택하여 진행한다.

### 2. Project Justification

- 1) Cost : 노동력과 서버비용 이외에 지출이 없어 저렴하다
- 2) Project Duration : 3 개월
- 3) Risk : 개발역량, 디자인 역량, 결석하고 나들이 가고 싶게 만드는 따스한 봄날씨, 막상 하는 것도 없으면서 괜히 들뜨는 축제기간, 암걸리는 신공학관 엘리베이터
- 4) Effect
  - 인천공항을 방문하는 여행객이 쉽게 환전을 처리할 수 있도록 돕는다.
  - 명세표를 문자로 확인할 수 있다.
  - 관리자가 쉽게 관리할 수 있도록 알림(기기오류/현금부족/현금포화)이 간다.

### 3. Risk Management

Risk	Probability	Significance	Weight
개발역량	5	10	50
디자인 역량	5	3	15
나들이 가고 싶게 만드는 따스한 봄 날씨	3	3	9
막상 하는 것도 없으면서 괜히 들뜨는 축제기간	4	7	28
암 걸리는 신공학관 엘리베이터	5	10	50

### 4. Risk Reduction plan

Risk	Way of Reduction
개발역량	구글링을 활용하거나 서적을 참고한다.
디자인 역량	디자인과 학생에게 꿀팁을 전수 받는다
나들이 가고 싶게 만드는 따스한 봄 날씨	어차피 나들이 갈 애인도 없다
막상 하는 것도 없으면서 괜히 들뜨는 축제기간	어차피 하는 것도 없는데 심장아 나대지마
암 걸리는 신공학관 엘리베이터	건대병원에서 암을 치료하는게 더 빠를듯하다

## 5. Market Analysis

기존 ATM 의 인터페이스는 버튼이 너무 많고 복잡하다. 인천공항에 환전 전용 ATM 이 존재하지만, 한국인을 초점으로 설계되어 외국인이 한화로 환전하기 어려운 구조이다

## 6. Other Managerial Issues

- 1)2018 년 6 월까지 Project 를 완성해야 한다.
- 2)추후 줄작으로 활용 가능

## Activity 1003. Define Requirements

### 1. Functional Requirements

No.	Function	Description
0	Read Item	카드 / 통장/관리자 카드 인식
1	Confirm	예금주 본인 확인
2	Check	계좌 잔액 확인
3	Deposit	현금을 받아, 확인하고, 계좌에 증감
4	Withdraw	고객이 원하는 현금만큼 계좌에서 차감하여 현금으로 교환
5	Transfer	고객이 원하는 금액만큼 입력한 타계좌로 이동
6	Update	거래 내역 업데이트
7	Receipt Print	모든 거래 이후 거래내역 출력
8	Status Alarm	ATM 현금 소지량 경고 알림
9	Cash Management	ATM 금고 관리

Ref.	Function	Category
R.0.0	Read Item	Evident
R.1.0	Confirm	Evident
R.1.1	Display Confirm Error	Evident
R.2.0	Check	Evident
R.2.1	Deposit	Evident
R.2.2	Withdraw	Evident
R.2.3	Transfer	Evident
R.3.0	Update	Hidden
R.3.1	Verify Sufficient Fund	Hidden

R.3.2	Receipt Print	Evident
R.4.0	Cash Management	Frill
R.4.1	Status Alarm	Hidden
R.4.2	Send Cash Saturation	Frill
R.4.3	Send Cash Starvation	Frill

## 2. Performance Requirements

- 외국인도 쉽게 이해할 수 있는 인터페이스를 가지고 있다.
- 고객이 정확하게 거래 내용을 확인할 수 있도록 한다.
- 최대한 간소화하여 고객이 거래에 집중할 수 있도록 한다.
- 적절한 본인 인증을 통하여 타의/악의적인 접근을 막는다.
- 기기는 충분한 현금을 소지하고 있다.

## 3. Operating Environments

- 1) OS : Window 7
- 2) 개발 언어 : JAVA

## 4. Development Environments

- 1) OS : Window 7, Mac OS
- 2) IDE : Eclipse
- 3) 개발 언어 : JAVA

## Activity 1004. Record Terms in Glossary

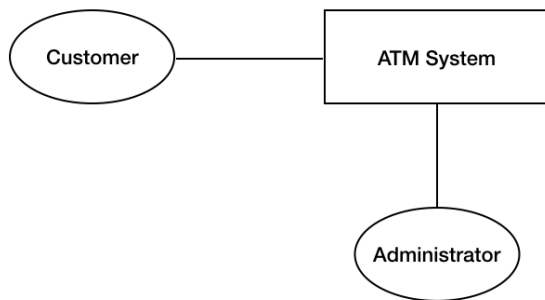
Function	Category
Item	카드 / 통장 / 관리자 카드
Confirm	본인 인증
Verify Sufficient Fund	충분한 잔액 확인
Check	잔고 확인
Deposit	입금
Withdraw	출금
Customer / User	고객
Transfer	송금
Receipt	명세표
Admin	ATM 관리자



Activity 1005.

-생략-

## Activity 1006. Define Draft System Architecture



## Activity 1007. Define Business Use Case

### 1 Determine System boundary



### 2 Identify and Describe Actors

#### 2.1 Customer

ATM System 을 예금조회, 인출, 송금, 입금 용도로 사용하는 객체

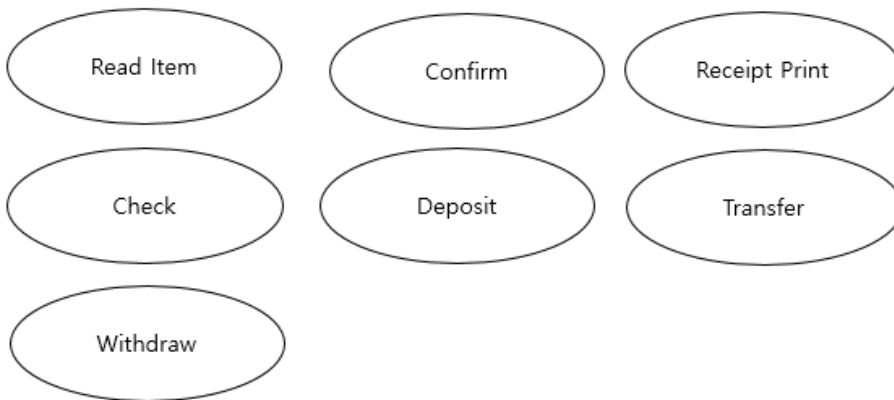
#### 2.2 Administrator

ATM 을 관리하는 객체

### 3 Identify Use-Case

#### 3.1 Use-case by actor-based

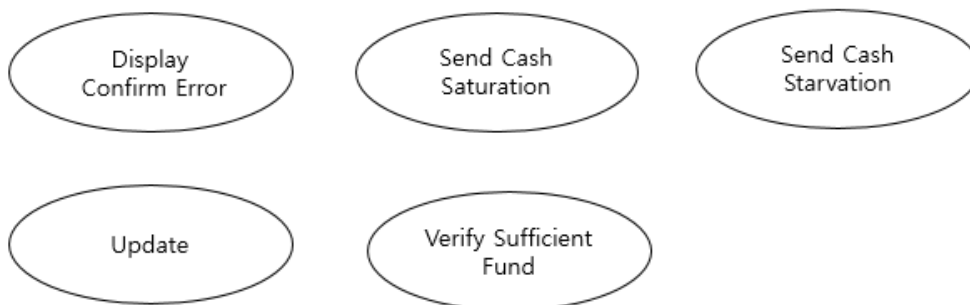
##### 3.1.1 Customer



##### 3.1.2 Administrator



##### 3.1.3 Use-case by event-based



### 4 Allocate system functions into Related Use-Cases

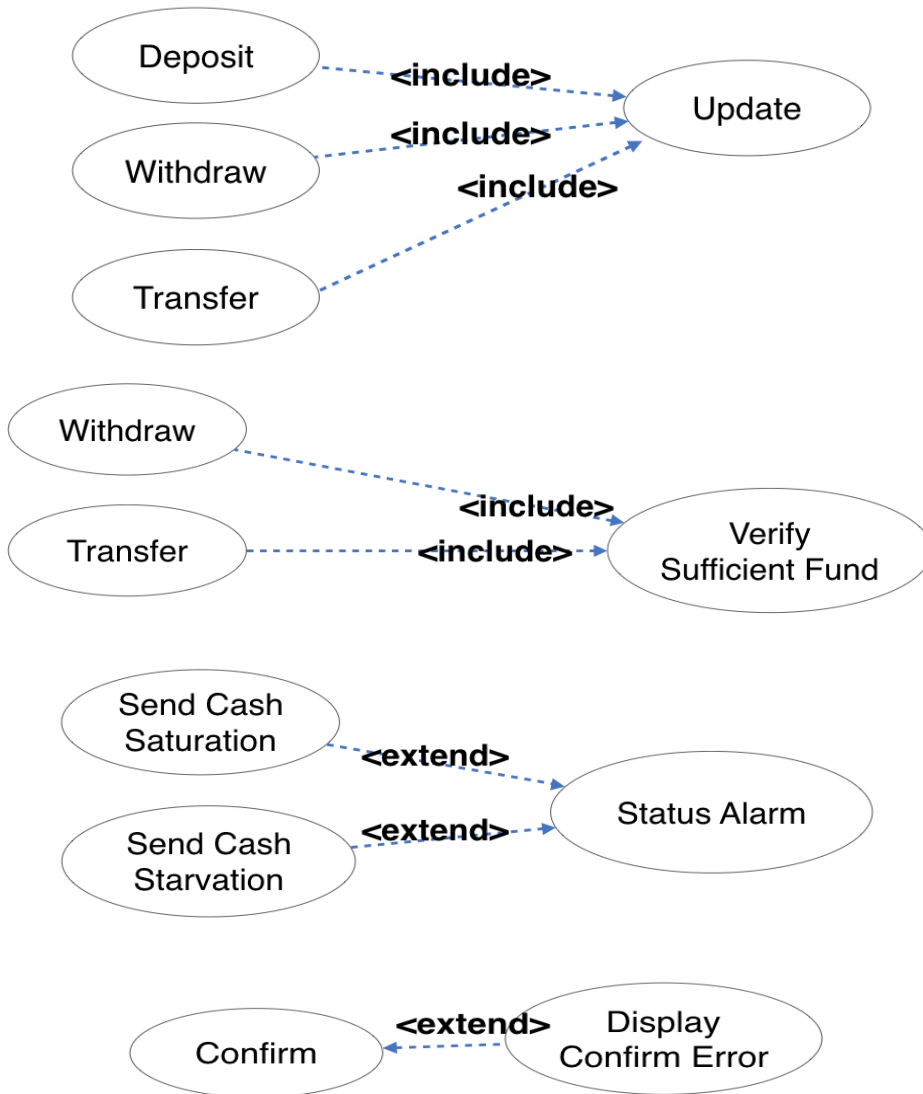
Ref.	Function	Use-Case
R.0.0	Read Item	Read Item
R.1.0	Confirm	Confirm
R.1.1	Display Confirm Error	Display Confirm Error
R.2.0	Check	Check
R.2.1	Deposit	Deposit
R.2.2	Withdraw	Withdraw
R.2.3	Transfer	Transfer
<b>R.3.0</b>	Update	Update
<b>R.3.1</b>	Verify Sufficient Fund	Verify Sufficient Fund
<b>R.3.1</b>	Receipt Print	Receipt Print
<b>R.4.0</b>	Cash Management	Cash Management
<b>R.4.1</b>	Status Alarm	Status Alarm
<b>R.4.2</b>	Send Cash Saturation	Send Cash Saturation
<b>R.4.3</b>	Send Cash Starvation	Send Cash Starvation

## 5 Categorize Use-Cases

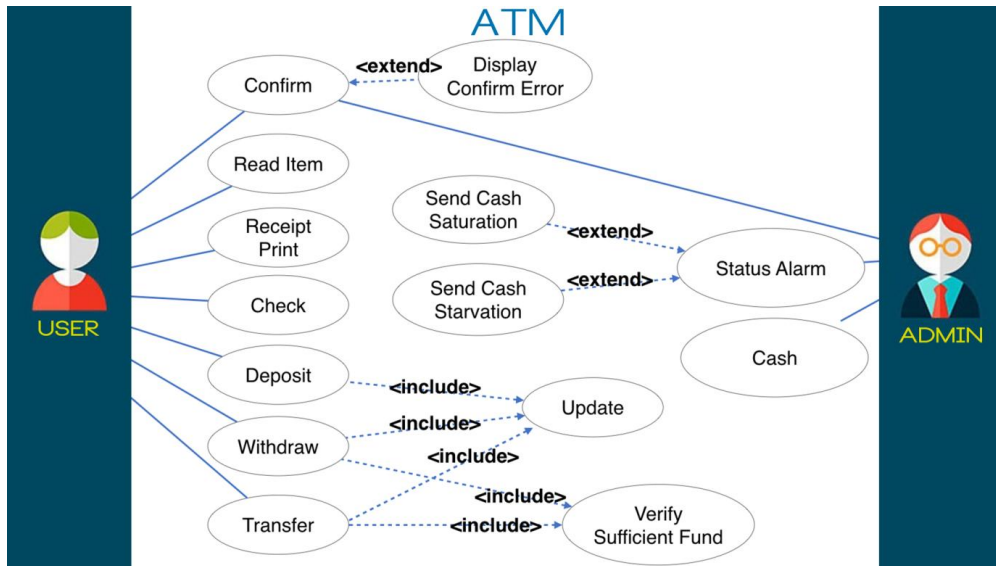
Use-case Number & Name	Category
<b>0. Read Item</b>	Primary
<b>1. Confirm</b>	Primary
<b>2. Display Confirm Error</b>	Secondary
<b>3. Check</b>	Primary
<b>4. Deposit</b>	Primary
<b>5. Withdraw</b>	Primary
<b>6. Transfer</b>	Primary
<b>7. Update</b>	Primary
<b>8. Verify Sufficient Fund</b>	Primary
<b>9. Receipt Print</b>	Primary
<b>10. Cash Management</b>	primary
<b>11. Status Alarm</b>	Secondary
<b>12. Send Cash Saturation</b>	Secondary
<b>13. Send Cash Starvation</b>	Secondary

**메모 [전1]:** Primary use cases : major common processes • Secondary use cases : minor or rare processes • Optional use cases : processes that may not be tackled

## 6 Identify the relationships between Use-Case



## 7 Draw a Use-Case diagram



## 8 Describe Use-Case

Use Case	1. Confirm
Actors	User
Description	Item 의 주인이 맞는지 확인한다

Use Case	2. Display Confirm Error
Actors	ATM
Description	Item 의 주인이 아니면 화면에 Error 문구를 출력한다

Use Case	3. Check
Actors	User
Description	User 의 계좌 잔고를 조회한다.

Use Case	4. Deposit
Actors	User
Description	User 의 계좌에 입금한다.

Use Case	5. Withdraw
Actors	User
Description	User 의 계좌로부터 출금한다.

Use Case	6. Transfer
Actors	User
Description	User 의 계좌에 있는 금액 중 일부를 원하는 계좌로 송금한다.

Use Case	7. Update
Actors	ATM
Description	거래 정보를 은행에 최신화시킨다.

Use Case	8. Verify Sufficient Fund
Actors	ATM
Description	User 의 계좌 잔고가 송금/출금 하기 원하는 금액보다 충분한지 확인한다.

Use Case	9. Receipt Print
Actors	User
Description	거래 내역을 출력한다.

Use Case	10. Cash Management
Actors	Admin
Description	ATM 내의 현금을 관리한다

Use Case	11. Status Alarm
Actors	Admin
Description	ATM 내의 정보를 Admin 에게 알려준다

Use Case	12. Send Cash Saturation
Actors	ATM
Description	ATM 내에 현금이 포화되었음을 알린다.

Use Case	13. Send Cash Starvation
Actors	ATM
Description	ATM 내에 현금이 부족함을 알린다.

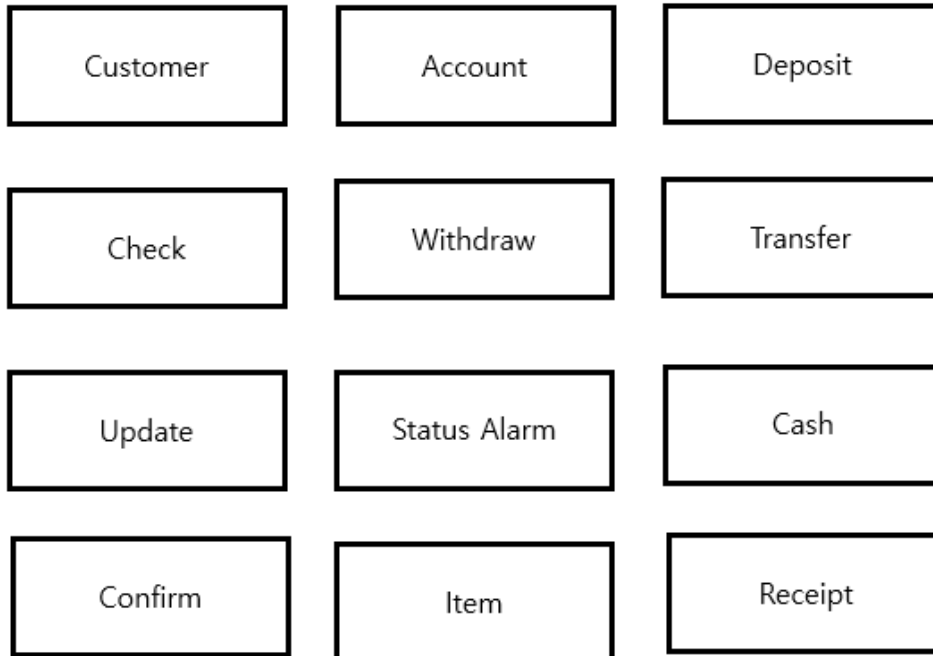
## 9 Rank Use-Case

Use-case Number & Name	Category
0. Read Item	Medium
1. Confirm	Medium
2. Display Confirm Error	High
3. Check	High
4. Deposit	High
5. Withdraw	High
6. Transfer	High
7. Update	High
8. Verify Sufficient Fund	Low
9. Receipt Print	Low
10. Cash Management	Low
11. Status Alarm	Low
12. Send Cash Saturation	Low
13. Send Cash Starvation	Low

메모 [전2]: High Buy Items It's the triggering event of all processes  
 Medium Add New Users Log In Refund Items Affects security ... ..  
 Low Cash out Start Up Shut Down Minimum effect on the architecture

이거 이제 읽어서  
 34567 High  
 01 Medium  
 28910111213 low

### Activity 1008. Define Business Concept Models



### Activity 1009. Test Case

Ref.	Function	Test Case
R.0.0	Read Item	- 적절한 Item 이 삽입되었는지 확인한다. - 삽입된 Item 이 유효한지 확인한다.
R.1.0	Confirm Owner	- Item 소유자가 맞는지 확인한다.
R.1.1	Display Confirm Error	-본인 인증이 잘 이루어지지 않았을 경우, 적절한 메시지를 출력하는지 확인한다.
R.2.0	Check	- 계좌의 잔고가 출력되는지 확인한다.
R.2.1	Deposit	- 고객이 넣은 돈이 입금되었는지 확인한다.
R.2.2	Withdraw	- 고객이 원하는 금액이 인출되었는지 확인한다.
R.2.3	Transfer	- 고객이 원하는 금액이 원하는 계좌로 송금되었는지 확인한다.
R.3.0	Update	- 거래 내역 최신화가 되어있는지 확인한다.



R.3.1	Verify Sufficient Fund	-요구한 금액만큼 통장잔고가 없는 경우 .인출/송금을 못하는지 확인한다. -통장잔고가 요구한 금액보다 많을 경우 인출/송금이 제대로 이루어 지는지 확인한다.
R.3.2	Receipt Print	-은행업무 후, 명세표 출력 요청 시 제대로 된 명세표가 출력되는지 확인한다.
R.4.0	Cash Management	- 관리자가 접근하여 ATM 속 지폐 수를 추가/감소 시킬 수 있는지 확인한다. -관리자가 아닌 사람이 접근하고자 할 경우 거부하는지 확인한다
R.4.1	Status Alarm	- 관리자에게 알림이 가는지 확인한다.
R.4.2	Send Cash Saturation	-ATM 안 현금이 수용할 수 없을 정도로 가까워져 갔을 경우, 관리자에게 알림이 가는지 확인한다.
R.4.3	Send Cash Starvation	-ATM 안 현금이 고객의 은행업무에 방해를 줄 정도로 적을 경우, 관리자에게 알림이 가는지 확인한다.

## Activity 1010. Refine Plan

### 1 Project Scope

지금 구축하는 ATM System 은 기존의 ATM 기기와 마찬가지로 예금조회, 인출, 입금, 송금기능을 수행하고 있습니다. 여기서 외화를 이용한 거래를 가능하게 함으로써 타 시스템과 경쟁력을 가질 수 있도록 합니다.

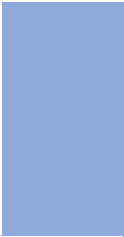

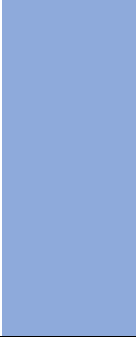
### 2 Project Objectives

Global ATM System 을 개발하여 기존 ATM 기기의 기능을 외화로도 가능하게 함으로써 여행객(외국/내국인)들에게 조금 더 편한 거래를 할 수 있도록 한다.

### 3 Functional Requirements

- Read Item
- Confirm
- Check
- Deposit
- Withdraw
- Transfer



2037. Define State Diagrams 2038. Refine System Test Case 2039. Analyze (2030) Traceability Analysis	
2040. Design 2041. Define Real Use Cases 2042. Define Reports, UI and Storyboards 2043. Refine System Architecture 2044. Define Interaction Class Diagrams 2045. Define Design Class Diagrams 2046. Define Database Schema 2047. Define Database Schema	
2050. Construct 2051. Implement Class & Interface Definition 2052. Implement Methods 2053. Implement Windows 2053. Implement Reports 2055. Write Unit Test Code 2060. Test 2061. Unit Testing 2062. Integration Testing 2063. System Testing 2064. Performance Testing 2065. Acceptance Testing 2066. Documentation Testing	

## 8. Configuration Management

사이트의 기본적인 기능이나 그 용어에는 변화가 없다.

## 9. Quality Assurance Plan

정해진 기간내에 요구하는 기능들을 수행한다