A New Reliable ATM

OOPT Stage 1000

Plan and Elaboration

Project Team T6

201411140 권성완
201511247 김선정
201510436 허윤아
201510285 조수빈

Date

2018-04-26

Activity 1001. Define Draft Plan

1. Motivation
   기술의 발전으로 인하여 현금보다 카드를 많이 사용하게 되어 사람들이 현금을 많이 갖고 다니지 않게 되었다. 하지만 급하게 현금이 필요한 경우가 여전히 종종 남아있어 ATM 기기의 필요성은 더욱 증대되어 이 프로그램을 개발하게 되었다. 이 시스템을 통하여 고객들은 현금 입출금 서비스 뿐만 아니라 대출, 공과금 납부, 환전 등 다양한 업무들을 은행을 가지 않고도 이용할 수 있게 될 것이다.

2. Project Objectives & Scope
   - 은행에 가지않고 간단한 은행 업무를 볼 수 있다.
   - 현금서비스를 이용할 수 있다.

3. Functional Requirements
   - Deposit
   - Withdraw
   - Transfer
   - Check Balance
   - Exchange
   - Loan
   - Pay Utility Bill
   - Take charge
   - Request Customer's Data
   - Check Validation
   - Check Password
   - Update Server Information
   - Print Error
   -

4. Non-functional Requirements
   - 24시간 동안 구동될 시스템이므로 안정적으로 설계한다.
   - 돈과 관련된 정보를 다루므로 정보가 정확해야 한다.

- 환전 시 환율은 국가마다 다르다.
- 환전 가능한 외화는 엔화, 위안화, 달러, 유로로 총 네 종류이다.
- 시스템에 대한 컴퓨터 상에서의 시뮬레이션이므로 모든 입력은 키보드를 통해 이루어진다.

5. Resource Estimation
   - Human Efforts(Man-Month) : 4 – 3months
   - Project Duration : 12 weeks
   - Other Efforts : 좋은 학점을 향한 집념, 조원들의 협동심
6. Other Information
   N/A

Activity 1002. Preliminary Investigation Report

1. Alternative Solutions

   - 기존의 잘 짜여진 시스템을 구매한다
   - 다른 소프트웨어 개발 회사에 의뢰한다.

2. Project Justification (Business Demands)

   - Cost : 시중의 제품보다 저렴하게 개발할 수 있다(인력비 감소).
   - Duration : 회사에 의뢰했을 때보다 짧은 시간 내에 개발이 가능하다.
   - Risk : 숙달되지 않은 개발자들의 개발로 인해 에러가 발생할 수 있고, 이로 인해 requirement를 충족시키지 못할 수 있다.
   - Effect : requirement를 직접 짜 구현할 수 있고, 노력으로 비용을 감소시킬 수 있다.

3. Risk Management

Risk	Probability	Significance	Weight
Lack of JAVA Experience	3	4	12
Lack of UML Experience	3	4	12
Lack of domain knowledge	3	4	12
Lack of GUI Experience	3	4	12
Lack of Time	2	2	4
Lack of Sleep	2	2	4

4. Risk Reduction plan

   - Lack of Java Experience (12) : JAVA 언어에 관한 책을 통해 공부한다.
   - Lack of UML Experience (12) : 강의시간에 배운 내용을 복습하고 팀원끼리 서로 알려준다.
   - Lack of domain knowledge (12) : domain diagram을 통해 domain에 대한 이해도를 늘린다. 평소 ATM기를 사용했던 경험을 떠올리며 작업한다.
   - Lack of GUI Experience (12) : 책과 우리의 친구 구글에게 도움을 요청한다.
   - Lack of Time (4) : 고학년들의 고질적인 문제로 부족한 시간은 여가시간 등을 줄여서 채운다.
   - Lack of sleep (4) : 어느정도 감수할 부분으로 개인이 체력관리에 힘쓴다.

5. Market Analysis

   - 이미 상용화되어 있는 ATM들이 많이 있으므로 이를 참조할 수 있으나, 개발의 필요성이 줄어든다.

   - 상용화되어 있는 ATM들의 기능이 다양하므로, 이를 차용한다.

6. Other Managerial Issues

   - 홈페이지에 기재된 개발 일정에 맞추어 작업한다.

Activity 1003. Define Requirements

1. Functional Requirements (Version 2.0)

- Deposit

  고객은 체크카드 또는 통장을 이용해 자신의 계좌에 현금을 입금한다. 신용카드를 통해 입금할 경우 이전에 대출(loan) 기록이 있을 때 해당 신용카드의 대출 금액에서 입금한 금액만큼 자동으로 갚아진다. ATM이 처리할 수 있는 금액의 단위는 만원, 5만원 단위이다.

- Withdraw

  고객은 카드 또는 통장을 이용해 자신의 계좌에서 현금을 인출한다. 출금 시 고객은 계좌의 비밀번호를 알고 있어야 한다. 비밀번호가 정확하지 않을 경우 거래는 진행되지 않는다. ATM이 처리할 수 있는 금액의 단위는 만원, 5만원 단위이다.

- Deposit without bankbook

  고객은 입금하려는 계좌번호를 정확히 알고 있을 경우 무통장 입금을 할 수 있다. 만약 계좌 번호가 정확하지 않을 경우, 거래는 진행되지 않는다. ATM이 처리할 수 있는 금액의 단위는 만원, 5만원 단위이다.

- Transfer

  고객은 카드 또는 통장을 이용해 자신의 계좌에서 돈을 송금할 수 있다. 송금 시에는 정확한 상대방의 계좌번호와, 자신의 계좌의 비밀번호를 알고 있어야 한다. 계좌번호와 비밀번호가 정확하지 않을 경우 거래는 진행되지 않는다.

- Check Balance

  고객은 카드 또는 통장을 이용해 해당 계좌의 최근 거래 내역과 각각의 거래 내역에 따른 잔고를 조회할 수 있다. 고객은 계좌의 비밀번호를 알고 있어야 한다. 조회 후 고객이 확인 버튼을 누르면 초기 화면으로 돌아간다.

- Exchange

  환율에 따라 고객이 원하는 원화에서 국가의 현금으로 환전해준다. 환율은 국가마다 상이하다. 환전 시 현금이 아닌, 고객 본인의 카드 또는 통장이 있어야 한다. ATM이 처리할 수 있는 금액의 단위는 만원, 5만원 단위이다.

- Loan

  고객의 신용카드 한도 내에서 현금을 인출 할 수 있는 기능으로, 고객은 신용카드의 비밀번호를 알고 있어야 한다. ATM이 처리할 수 있는 금액의 단위는 만원 단위이다.

- Pay Utility Bill

  고객은 지로 고지서를 입력해, 체크카드 혹은 통장을 통해 공과금을 납부한다. 고객은 계좌의 비밀번호를 알고 있어야 한다.

- Print Transaction Receipt

  모든 process를 종료한 이후 서버를 업데이트하기 전에 거래 명세서를 출력한다.

- Take Charge

  환전, 타행 송금, 신용카드를 이용한 현금 출금을 할 때 발생하는 수수료를 받는 행위로 등급에 따라 면제될 수 있다.

- Request Customer's Data

  ATM은 Offer 서버에 저장되어 있는 체크카드 정보, 통장 정보, 신용카드, 계좌 정보를 요청한다.

- Check Validation

  고객은 카드 또는 통장을 투입하고, 거래를 시작한다. ATM은 offer에게 고객의 정보를 요청하여 투입된 카드 또는 통장의 정보가 유효한지 확인한다.

- Check Password

  ATM은 Offer 서버에 정보를 요청해, 고객이 입력한 계좌 또는 신용카드의 비밀번호가 기존에 등록된 것과 일치하는지 확인한다. 비밀번호 오류가 3회 이상 누적될 경우 고객은 해당 계좌 혹은 카드로 ATM을 이용할 수 없다.

- Update Server Information

  ATM은 잔고 확인을 제외한 모든 process를 종료한 이후, 추가 혹은 변경된 정보를 갱신할 것을 Offer 서버에 요청한다.

- Print Error

  ATM은 Error가 발생한 상황에서 Error 메시지를 고객에게 알린다. Error가 발생하는 상황은 고객의 정보가 서버에 등록된 정보와 일치하지 않을 때이다.

- Proceed Forced Termination

  비밀번호 오류를 제외한 오류가 3번보다 더 많이 발생할 경우 강제로 process를 종료하고 기본 화면으로 돌아간다.

- Transaction Lock

비밀번호 오류가 3회 이상 누적된 고객의 경우, 부정한 거래 시도로 간주되어 ATM 이용이 금지된다.

- Check Credit

  고객의 신용 등급을 조회한다.

2. System Functions (Table)

Ref. #	Function	Category
R1.1	Deposit	Evident
R1.2	Withdraw	Evident
R1.3	Deposit Without bankbook	Evident
R1.4	Transfer	Evident
R1.5	Exchange	Evident
R1.6	Loan	Evident
R1.7	Check Balance	Evident
R1.8	Pay Utility Bill	Evident
R2.1	Print Transaction Receipt	Evident
R2.2	Take Charge	Evident
R2.3	Print Error	Evident
R2.4	Proceed Forced Termination	Evident
R2.5	Transaction Lock	Evident
R3.1	Request Customer's Data	Hidden
R3.2	Check Validation	Hidden
R3.3	Check Password	Hidden
R3.4	Check Credit	Hidden
R4.1	Update Server Information	Hidden

3. Performance Requirements

   - 현금 거래에 관한 정보를 신속하고 정확하게 처리하여 서비스를 제공한다.

4. Operating Environments

   - OS : Microsoft Windows 10 (64bit)
   - CPU : i5-7200U
   - Memory : 8GB

5. Development Environments

- OS : Microsoft Windows 10 pro (64bit)
- CPU : i5-7200U
- Memory : 8GB
- Program Language : JAVA
- Editor : eclipse

6. Interface Requirements
   - 사용자의 메뉴 선택에 의해 각 기능들을 수행할 수 있는 화면으로 전환된다.
   - 다음 버전의 업데이트를 통해 좀 더 정교한 프로그램을 설계한다.
7. Other Requirements

   (N/A)

Activity 1004. Record terms in Glossary

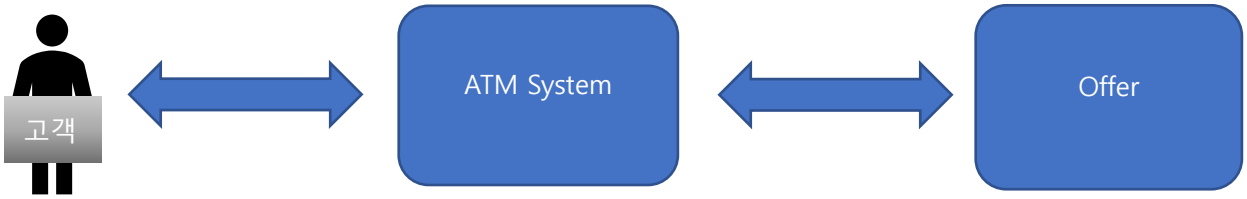
Term	Description	Remarks
ATM	Automated Teller Machine의 준말로 현금 자동 입출금기를 의미한다	
Offer	ATM 서비스를 제공하는 회사로 은행과 카드사가 이에 해당한다.	
Bankbook	고객의 계좌 정보를 담고있는 통장	
DWB	Deposit without bankbook의 약자로, 무통장 입금을 의미한다	
KRW	Korean Won, 원화 단위를 의미한다	
Giro bill	지로 고지서를 의미한다	
Credit rating	신용 등급을 의미한다	

Activity 1005. Implement Prototype

(생략)

Activity 1006. Define Business Use-Cases

1. Define System Boundary



2. Identify and Describe Actors

   - Customer : ATM을 이용하는 고객
   - ATM : Offer와 고객의 중간다리 역할을 해주는 시스템. 고객이 더 편하게 금융거래를 할 수 있도록 돕는다.
   - Offer(Secondary Actor) : 고객의 금융거래 정보를 저장하고 제공하는 secondary actor

3. Identify Use-Case

   Use Cases by Actor – Customer

Deposit	Withdraw	transfer	check balance	Deposit without bankbook
Exchange	Insert	loan	pay utility bill	

   Use Cases by Actor – Offer

Update Server System

Use Cases by Event

Request Customer's Data	Print Error Message	Transaction Lock	Check Validation
pay utility bill	Take charge	Print transaction receipt	Proceed Forced Termination
Check Credit	Check password		

4. Allocate system functions into Related Use Cases

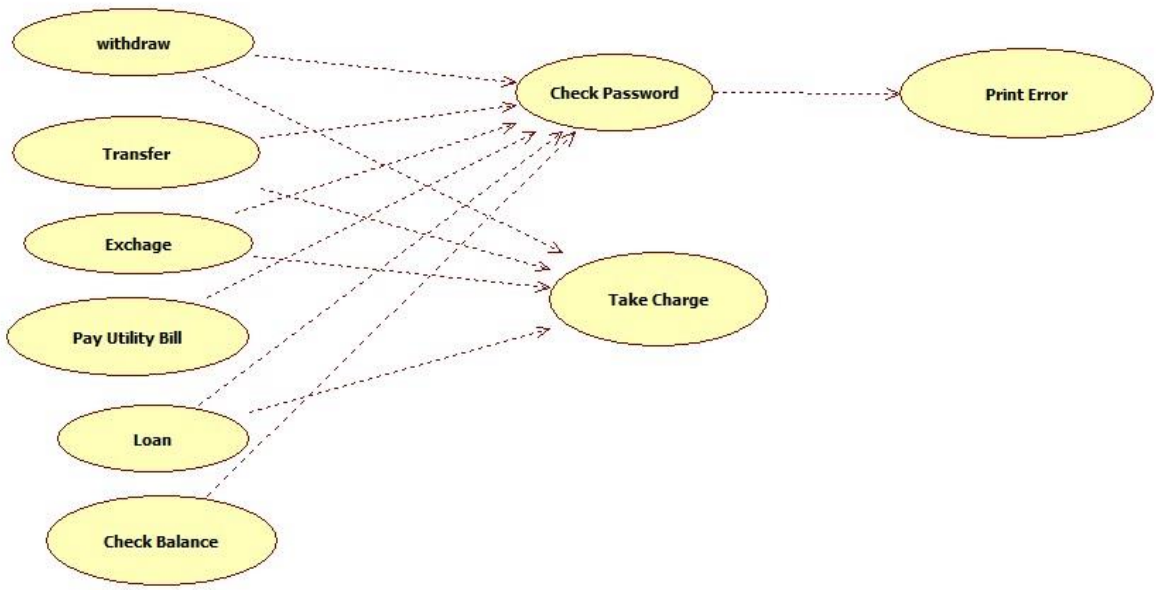
Ref. #	Function	Use Case Number & Name	Category
R1.1	Deposit	1. Deposit	
R1.2	Withdraw	2. Withdraw	
R1.3	Deposit Without bankbook	3. Deposit without bankbook	
R1.4	Transfer	4. Transfer	
R1.5	Exchange	5. Exchange	
R1.6	Loan	6. loan	
R1.7	Check Balance	7. Check balance	
R1.8	Pay Utility Bill	8. Pay utility bill	
R2.1	Print Transaction Receipt	9. Print Transaction Receipt	
R2.2	Take Charge	10. Take Charge	
R2.3	Print Error	11. Print Error	
R2.4	Proceed Forced Termination	12. Forced Termination	
R2.5	Transaction Lock	13. Transaction Lock	
R3.1	Request Customer's Data	14. Request customer's data	
R3.2	Check Validation	15. Insert	
R3.3	Check Password	16. Check Password	
R3.4	Check Credit	17. Check Credit	
R4.1	Update Server Information	18. Update Server Information	

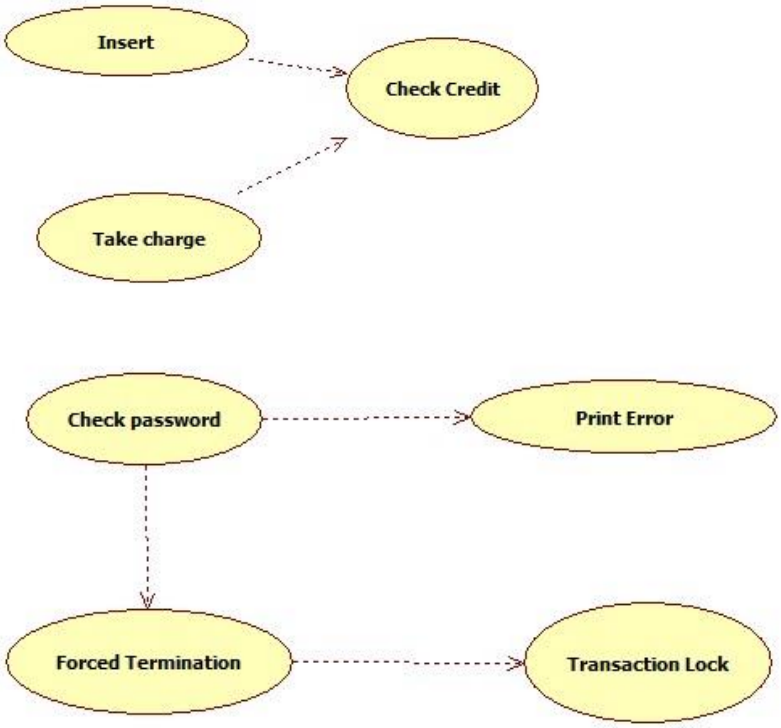
5. Categorize Use Cases

Ref. #	Function	Use Case Number & Name	Category
R1.1	Deposit	1. Deposit	Primary
R1.2	Withdraw	2. Withdraw	Primary

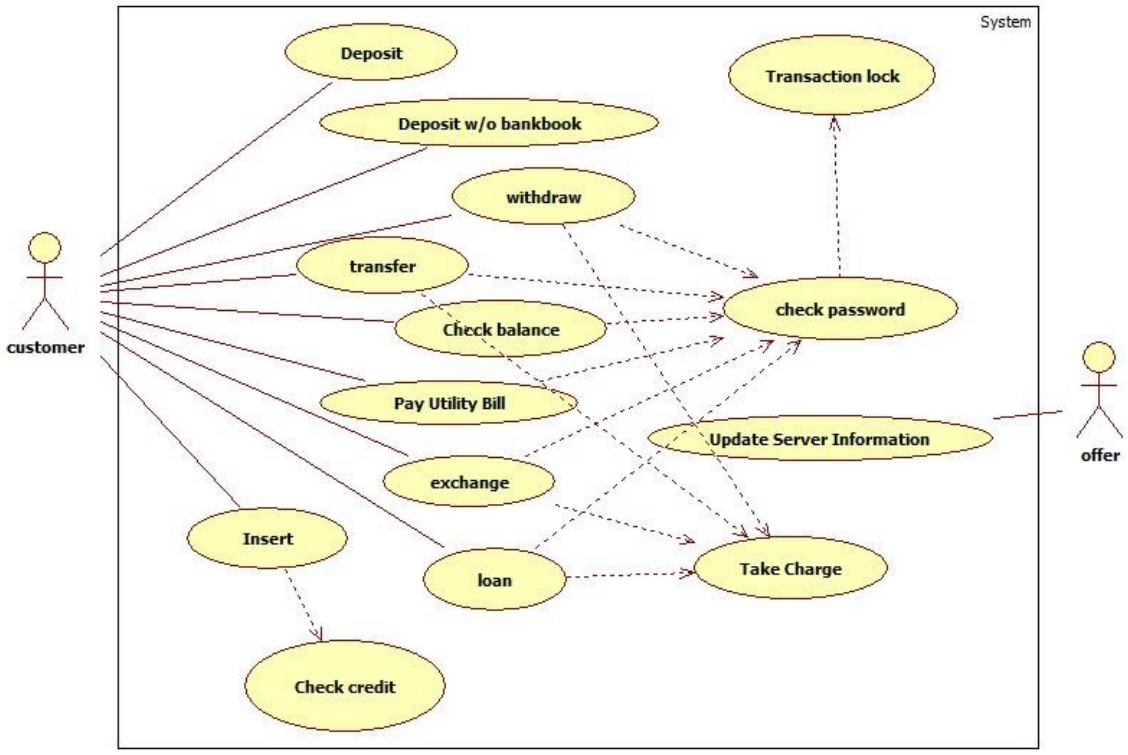
R1.3	Deposit Without bankbook	3. Deposit without bankbook	Primary
R1.4	Transfer	4. Transfer	Primary
R1.5	Exchange	5. Exchange	Primary
R1.6	Loan	6. loan	Primary
R1.7	Check Balance	7. Check balance	Primary
R1.8	Pay Utility Bill	8. Pay utility bill	Primary
R2.1	Print Transaction Receipt	9. Print Transaction Receipt	Primary
R2.2	Take Charge	10. Take Charge	Secondary
R2.3	Print Error	11. Print Error	Primary
R2.4	Proceed Forced Termination	12. Forced Termination	Primary
R2.5	Transaction Lock	13. Transaction Lock	Primary
R3.1	Request Customer's Data	14. Request customer's data	Primary
R3.2	Check Validation	15. Insert	Primary
R3.3	Check Password	16. Check Password	Primary
R3.4	Check Credit	17. Check Credit	Primary
R4.1	Update Server Information	18. Update Server Information	Primary

6. Identify the relationships between Use-Case





7. Draw a Use-Case diagram



8. Describe Use-Case

Use Case	1. Deposit
Actors	Customer
Description	-Customer can deposit cash into account by using card or bankbook. -Customer can deposit money into credit card. -Customer can only deposit money in unit of 10000₩, 50000₩. -If customer successfully deposits, ATM requests Offer to update server information.(Hidden) -If customer deposits cash into credit card, ATM requests Offer to update server information. Amount of money customer inputted is automatically used to repay from the amount customer loaned. (Hidden)

Use Case	2. Withdraw
Actors	Customer
Description	-Customer can withdraw cash from account by using card or bankbook. -Customer can only withdraw money in unit of 10000₩ and 50000₩ under balance. -Customer has withdrawing limit at a day. -Customer needs to choose the number of 50000₩. -Customer needs to know password of an account. -If customer successfully withdraws, ATM requests Offer to update server information.(Hidden)

Use Case	3. Deposit Without bankbook
Actors	Customer
Description	-Customer can transfer money to other's account by paying cash. -Customer needs to input receiver's account number correctly. -If customer did it wrong, ATM prints out the error message.

Use Case	4. Transfer
Actors	Customer
Description	-Customer can transfer money to other's account by using check card or bankbook.

	-If customer uses check card or bankbook, ATM requests password and balance to Offer's server. -If customer enters wrong password, ATM prints out error message. -Customer needs to input receiver's account number correctly. -If customer did it wrong, ATM prints out the error message. -If customer is transferring money to other bank's account, ATM takes charge of certain percentage. -If customer successfully transfers money from account ATM requests Offer to update server information. (Hidden)

Use Case	5. Exchange
Actors	Customer
Description	-Customer can exchange KRW into foreign currency. -Customer can only use check card or bankbook to exchange money. -If customer enters wrong password, ATM prints out error message. -If customer successfully exchanges money, ATM requests Offer to update server information. (Hidden)

Use Case	6. Loan
Actors	Customer
Description	-Customer can withdraw cash in unit of 10000₩ and 50000₩ with credit card, and we call it Loan. -Each customer has loan limits. -If customer enters wrong password, ATM prints out error message. -Credit Card Company(included in Secondary Actor called Offer) lends money, and requests ATM to take charge. -If customer successfully loans money, ATM requests Offer to update server information. (Hidden)

Use Case	7. Check balance
Actors	Customer
Description	-Customer can check balance of own account. -If customer enters wrong password, ATM prints out error message. -ATM prints out customer's recent transaction history. -If customer inputs 'OK' button, ATM returns to basic screen.(GUI level)

Use Case	8. Pay Utility Bill
Actors	Customer
Description	-Customer can only use check card or bankbook to pay utility bill. -Customer needs to input giro bill before paying bill. -Customer can only deposit money into virtual payment account. -If customer enters wrong password, ATM prints out error message.

Use Case	9. Print Transaction Receipt
Actors	(None)
Description	-ATM prints out transaction receipt after end of every process. -Transaction Receipt has record of current transaction such as amount of money used in transaction, account balance or etc.

Use Case	10. Take Charge
Actors	(None)
Description	-ATM takes charge from customer in process of transfer, loan, and exchange. -If customer's credit rating is high, ATM doesn't take charge.

Use Case	11. Print Error
Actors	(None)
Description	-This use case begins when customer inputted wrong information except password, such as account number, card, bankbook, giro bill, or when customer wants to withdraw or loan or deposit and something went wrong. -If the information customer inputted is different from the information from the Offer's server data, or if the balance of an account or loan limit is not enough, or if cash customer dr or withdraws is not in unit of 10000₩ and 50000₩, ATM prints out error message to customer.

Use Case	12. Forced Termination
Actors	(None)
Description	-This use case begins when customer inputted wrong information except password over 3 times. -Every process is immediately stopped and returns to basic

	screen.(GUI level)

Use Case	13. Transaction Lock
Actors	(None)
Description	-This use case begins when customer inputted wrong password for account or credit card over 3 times. -Customer cannot use account or credit card immediately, and cannot unlock with ATM.

Use Case	14. Request customer's data
Actors	(None)
Description	-This use case begins when customer inputs information such as card, bankbook, account number, giro bill, or when customer wants to withdraw or loan. -ATM requests customer's data to Offer's server, and compares to check error.

Use Case	15. Insert
Actors	Customer
Description	-This use case begins when customer choose menu from the basic screen in GUI level, except DWB. -When customer inputs card or bankbook, ATM requests customer's data to the Offer's server to compare information inputted.

Use Case	16. Check password
Actors	(None)
Description	-This use case begins when customer inputs password of an account or credit card in every process except deposit and DWB. -ATM compares data requested to Offer's server.

Use Case	17. Check Credit
Actors	(None)
Description	-This use case begins when ATM needs to take charge from customer. -ATM request Offer to provide customer's credit rating. -If customer's credit rating is high, ATM doesn't take charge.

Use Case	18. Update Server Information
Actors	Offer
Description	-This use case begins when every process except check balance ends. -ATM requests to update Offer's server data by sending updated data(recent transaction records). -Offer update information provided by ATM.

9. Rank Use-Cases

Ref. #	Function	Use Case Number & Name	Category	Rank
R1.1	Deposit	1. Deposit	Primary	High
R1.2	Withdraw	2. Withdraw	Primary	High
R1.3	Deposit Without bankbook	3. Deposit without bankbook	Primary	High
R1.4	Transfer	4. Transfer	Primary	High
R1.5	Exchange	5. Exchange	Primary	High
R1.6	Loan	6. loan	Primary	High
R1.7	Check Balance	7. Check balance	Primary	High
R1.8	Pay Utility Bill	8. Pay utility bill	Primary	High
R2.1	Print Transaction Receipt	9. Print Transaction Receipt	Primary	High
R2.2	Take Charge	10. Take Charge	Secondary	Medium
R2.3	Print Error	11. Print Error	Primary	High
R2.4	Proceed Forced Termination	12. Forced Termination	Primary	High
R2.5	Transaction Lock	13. Transaction Lock	Primary	High
R3.1	Request Customer's Data	14. Request customer's data	Primary	High
R3.2	Check Validation	15. Insert	Primary	High
R3.3	Check Password	16. Check Password	Primary	High
R3.4	Check Credit	17. Check Credit	Primary	High
R4.1	Update Server Information	18. Update Server Information	Primary	High

Activity 1007. Define Business Concept Model

Customer(User)	ATM	Credit Card Company	Bank	Credit Card	Check Card	Cash
Name	Password	Utility Bill	Exchange rate(USD)	Exchange rate(JPY)	Exchange rate(EUR)	Exchange rate(CNY)
Deposit	Withdraw	Transfer	Check Balance	Exchange	Loan	Input
Commission	Store	Check Error	Print Error	Bank book	Limit	Dept
Account Number	Account Owner	10000₩	50000₩	Grade	Customer's Bank Data	Customer's Grade Data

Activity 1008. Define System Test Case

Ref. #	Function	Use Case Name	Test	Test description
R1.1	Deposit	1. Deposit	계좌에 맞게 돈이 입금되는지 TEST.	- 통장/체크카드 이용 시 해당 계좌에 맞게 입금되는지 TEST - 만원 단위로 입금되는지 TEST - 신용카드를 이용할 시, 대출금이 상환 되는지 TEST
R1.2	Withdraw	2. Withdraw	고객의 계좌에서 인출이 되는지 TEST	- 비밀번호가 맞아야 인출되는지 TEST - 만원 단위로만 인출되는지 TEST - 고객이 선택한 오만원권과 만원권의 장 수대로 출금되는지 TEST - 인출 제한 금액을 넘지 않았는지 TEST
R1.3	Deposit Without bankbook	3. Deposit without bankbook	무통장입금이 되는지 TEST	- 통장/체크카드 없이도 다른 사람에게 송금할 수 있는지 TEST - ATM과 같은 은행에만 보낼 수 있는지 TEST - 수신인의 정보가 정확해야 송금이 되는지 TEST - 만원 단위로 입금하는지 TEST - 수신자의 계좌에 돈이 입금되었는지 TEST
R1.4	Transfer	4. Transfer	송금이 되는지 TEST	- 고객이 통장이나 체크카드를 이용해서 돈을 보낼 수 있는지 TEST - 고객의 계좌에서 송금하는 금액 + (타행의 경우)수수료가 인출되는

				지 TEST - 수신자의 계좌에 돈이 입금되었는지 TEST - 통장/체크카드 이용 시 비밀번호가 맞는 경우만 돈을 보내는지 TEST - 수신인의 정보가 정확해야 송금이 되는지 TEST
R1.5	Exchange	5. Exchange	한화를 외화로 환전이 되는지 TEST	- 고객이 통장이나 체크카드를 입금했을 때만 환전이 가능한지 TEST - 비밀번호가 맞아야 환전이 되는지 TEST - 나라마다 환율이 맞게 적용되는지 TEST - 고객의 계좌에서 환전하는 금액 + 수수료만큼 인출되는지 TEST - 환전한 금액만큼 현금으로 나오는지 TEST
R1.6	Loanq	6. loan	대출(카드론)이 되는지 TEST	- 신용카드를 입금했을때만 대출이 되는지 TEST - 만원 단위로만 인출되는지 TEST - 한도를 넘지 않았을 경우에만 대출이 되는지 TEST - 비밀번호가 맞을 경우에만 대출이 되는지 TEST - offer(카드사)에 수수료+ 빌린 금액만큼 정보 업데이트 요청이 되는지 TEST
R1.7	Check Balance	7. Check balance	계좌조회가 되는지 TEST	- 통장이나 체크카드를 넣었을 때 계좌 조회를 할 수 있는지 TEST - 비밀번호가 맞을 경우에만 계좌조회가 되는지 TEST - 계좌의 최근 거래내역을 출력하는지 TEST
R1.8	Pay Utility Bill	8. Pay utility bill	공과금을 납부할 수 있는지 TEST	- 통장이나 체크카드를 넣었을 때 공과금을 납부 할 수 있는지 TEST - 지로를 통해 납부할 수 있는지 TEST - 가상계좌를 이용해 납부할 수 있는지 TEST - 비밀번호가 맞아야 납부할 수 있는지 TEST
R2.1	Print	9. Print	거래명세서 출력	- 각각의 프로세스가 끝난 후 거래

	Transaction Receipt	Transaction Receipt	이 되는지 TEST	명세서를 출력할 수 있는지 TEST - 입금; 거래명세서에 계좌정보/ 입금액/ 입금 후 금액 - 출금: 거래명세서에 계좌정보/ 출금액/ 출금 후 금액 - 송금: 거래명세서에 계좌정보/ 송금액/ 송금 후 금액 - 무통장입금: 거래명세서에 계좌정보/ 송금액 - 공과금 납부: 거래명세서에 계좌정보/ 납부액/ 납부 후 금액 - 환전: 거래명세서에 계좌정보/ 환전액/환전 후 금액 - 대출: 거래명세서에 카드정보/ 대출한 금액/ 갚아야 하는 금액(대출금+수수료) - 각각 맞게 출력되는지 TEST
R2.2	Take Charge	10. Take Charge	수수료가 부과되는지 TEST	- 송금/대출/환전 시 수수료가 붙는지 TEST - atm이 작업할 때, offer가 요청한 수수료만큼 더 붙여서 작업하는지 TEST - 고객의 신용등급이 높으면 atm이 수수료를 받지 않는지 TEST
R2.3	Print Error	11. Print Error	에러를 프린트하는지 TEST	- 비밀번호가 틀린 경우를 제외한 에러들을 프린트하는지 TEST - 각 과정에서 계좌/카드/체크카드/지로의 정보가 틀린 경우 에러출력 TEST - 출금/대출시 한도를 넘은 경우 에러 출력 TEST
R2.4	Proceed Forced Termination	12. Forced Termination	강제종료 TEST	- 비밀번호를 3번 틀렸을 경우 강제종료가 되는지(처음 화면으로 돌아가는지) TEST
R2.5	Transaction Lock	13. Transaction Lock	거래 잠금 TEST	- 비밀번호가 3번 틀렸을 시 해당 계좌/카드로 거래를 못 하게 잠기는지 TEST
R3.1	Request Customer's Data	14. Request customer's data	고객정보 얻어오기 TEST	- ATM이 OFFER에게서 고객정보를 얻어올 수 있는지 TEST - 각 과정에서 필요한 계좌/체크카드/카드/지로 등의 정보를 얻어오는지 TEST - 필요한 과정: 입금/ 출금/ 송금/

				대출/ 무통장입금/ 공과금 납부/ 환전
R3.2	Check Validation	15. Insert	통장/카드/체크카드 삽입 TEST	- 삽입된 매체(통장/카드/체크카드)가 유효한지 확인하는 기능 TEST
R3.3	Check Password	16. Check Password	비밀번호 확인 TEST	- 각 프로세스에서 입력한 비밀번호가 맞는지 확인하는 기능 TEST - 필요한 프로세스; 출금/ 송금/ 대출/ 계좌조회/ 공과금 납부/ 환전
R3.4	Check Credit	17. Check Credit	신용등급 확인 TEST	- ATM이 수수료를 부과할 때 신용등급을 확인 할 수 있는지 TEST
R4.1	Update Server Information	18. Update Server Information	서버 정보 업데이트 TEST	- 계좌조회를 제외한 각 프로세스에서, 프로세스가 끝난 후 서버가 정보를 업데이트 하는지 TEST - ATM이 서버에 정보 업데이트를 요청하면, 서버가 그대로 수행하는지 TEST - 입금; 입금 계좌/ 입금액/ 입금 후 금액 - 입금(카드사용); 카드 정보/ 상환액/ 상환 후 남은 대출금 - 출금; 출금 계좌/ 출금액/ 출금 후 금액 - 송금; 송금 계좌/ 송금액/ 송금 후 금액 - 송금; 송금 받는 계좌/ 송금액/ 송금 받은 후 금액 - 무통장입금; 송금 받는 계좌/송금액/ 송금 받은 후 금액 - 공과금 납부; 공과금이 인출된 계좌/ 인출액/ 인출 후 금액 - 환전; 환전 계좌/ 환전액/ 환전 후 금액(환전액+수수료) - 대출/ 카드 정보/ 대출 후 한도/ 대출금/ 상환해야 하는 금액(대출금+수수료)

Activity 1009. Refine Plan

1. Project Scope & Objectives

   - 은행에 가지않고 간단한 은행 업무를 볼 수 있다.
   - 현금서비스를 이용할 수 있다.

2. Functional Requirements

Ref. #	Function	Category
R1.1	Deposit	Evident
R1.2	Withdraw	Evident
R1.3	Deposit Without bankbook	Evident
R1.4	Transfer	Evident
R1.5	Exchange	Evident
R1.6	Loan	Evident
R1.7	Check Balance	Evident
R1.8	Pay Utility Bill	Evident
R2.1	Print Transaction Receipt	Evident
R2.2	Take Charge	Evident
R2.3	Print Error	Evident
R2.4	Proceed Forced Termination	Evident
R2.5	Transaction Lock	Evident
R3.1	Request Customer's Data	Hidden
R3.2	Check Validation	Hidden
R3.3	Check Password	Hidden
R3.4	Check Credit	Hidden
R4.1	Update Server Information	Hidden

3. Performance Requirements

- 현금 거래에 관한 정보를 신속하고 정확하게 처리하여 서비스를 제공한다.

4. Operating Environments

- OS : Microsoft Windows 10 (64bit)
- CPU : i5-7200U
- Memory : 8GB

5. User Interface Requirements

- 사용자의 메뉴 선택에 의해 각 기능들을 수행할 수 있는 화면으로 전환된다.
- 다음 버전의 업데이트를 통해 좀 더 정교한 프로그램을 설계한다.

6. Other Requirements (N/A)

7. Resources

- Human Efforts(Man-Month) : 4 – 3months
- Project Duration : 12 weeks
- Other Efforts : 좋은 학점을 향한 집념, 조원들의 협동심

8. Scheduling

Stage	Phase(00x0)/Activity(000x)	Schedule(Day) 1	2	3	4	5
1000. Plan & Elaborate	1001. Define Draft Plan 1002. Create Preliminary Investigation Report 1003. Define Requirements 1004. Record Terms in Glossary 1005. Implement Prototype 1006. Define Use Cases 1007. Define Draft Conceptual Model 1008. Define Draft System Architecture 1009. Refine Plan					
2000. Build	2010. Revise Plan 2020. Synchronize Artifacts 2030. Analyze 2031. Define Essential Use Case 2032. Refine Use Case Diagrams 2033. Refine Conceptual Model 2034. Refine Glossary 2035. Define System Sequence Diagrams 2036. Define Operation Contracts 2037. Define State Diagrams 2040. Design 2041. Define Real Use Cases 2042. Define Reports, UI and Storyboards 2043. Refine System Architecture 2044. Define Interaction Diagrams 2045. Define Design Class Diagrams 2046. Define Database Schema 2050. Construct 2051. Implement Class & Interface Definition 2052. Implement Methods. 2053. Implement Windows 2054. Implement Reports 2055. Implement DB Schema 2056. Write Test Code 2060. Test 2061. Unit Testing 2062. Integration Testing 2063. System Testing 2064. Performance Testing 2065. Acceptance Testing 2066. Documentation Testing					
3000. Deploy -ment	3001. Complete Technical Documents 3002. Complete User Documents 3003. System Testing 3004. Acceptance Testing 3005. Documentation Testing 3006. Train 3007. Establish Parallel Runs and Crossover 3008. Establish Support 3009. Install					

(SMA Case Study : Plan 마지막 도표 참조)