
Software Modeling & Analysis

[OOPT stage 2040]

No.	T7
Subject	Software Modeling & Analysis
Professor	JUNBEOM YOO
Team Member	201615007 문기태 201410621 한상민

Chart.

Activity 2041. Design Real Use Cases

Activity 2042. Define Reports & UI, and Storyboards

Activity 2043. Refine System Architecture

Activity 2044. Define Interaction Diagrams

Activity 2045. Define Design Class Diagrams

Activity 2046. Design Traceability Analysis

Activity 2041. Design Real Use Cases

Use Case	1. Send
Actor	User
Purpose	송금기능을 수행한다.
Overview	사용자가 원하는 대상의 계좌로 돈을 이체한다.
Type	Primary and essential
Cross Reference	System function: 1.2, 1.3 Use Case: Limited Amount, Check Password
Pre-Requisites	사용자와 송금대상의 계좌정보가 모두 txt파일에 있어야 한다.
Typical Courses of Events	(A): User (S): System 1. (A)가 화면에 있는 빈칸에 계좌번호를 입력한다. 2. (S)가 입력된 계좌번호에 해당되는 정보를 불러온다. 3. (A)가 송금 기능의 버튼을 누르고 비밀번호를 입력한다. 4. (A)는 상대 은행을 누르고 계좌번호를 입력한다. 5. (A)가 송금할 금액을 입력한다 6. (S)가 금액이 한도에 맞는지 확인하고 진행시킨다. 7. (S) 송금을 진행시키고 명세서를 출력할 것인지 물어본다. 8. (A) 명세서를 뽑을지 말지 Y/N을 누른다. 9. (S) 초기화면으로 되돌린다
Alternative Course of Events	N/A
Exceptional Courses of Events	(A)가 ATM기기에 걸려있는 한도금액보다 초과하여 입력하면 Limit amount에 의해 다시 인출 액 입력으로 돌아간다. (A)가 비밀번호를 잘못 입력하면 다시 비밀번호 입력으로 돌아가게 된다.

Use Case	2. withdraw
Actor	User
Purpose	인출기능을 수행한다.
Overview	사용자가 ATM기기에서 돈을 인출한다
Type	Primary and essential
Cross Reference	System function: 1.2, 1.3 Use Case: Limited Amount, Check Password
Pre-Requisites	해당 계좌의 정보가 txt파일에 있어야 한다.
Typical Courses of Events	(A): User (S): System 1. (A)가 화면에 있는 빈칸에 계좌번호를 입력한다. 2. (S)가 입력된 계좌번호에 해당되는 정보를 불러온다. 3. (A)가 인출 버튼을 누르고 인출할 금액을 입력한다

	<p>4. (S)가 금액이 한도에 맞는지 확인하고 비밀번호를 요구한다</p> <p>5. (A)가 비밀번호를 입력한다</p> <p>6. (S)가 비밀번호를 확인하고 인출을 시키며 명세서를 뽑을지 말지 물어본다</p> <p>7. (A)가 Y/N을 눌러서 명세서를 뽑을지 말지 결정한다</p> <p>8. (S) 초기화면으로 되돌린다</p>
Alternative Course of Events	N/A
Exceptional Courses of Events	<p>(A)가 ATM기기에 걸려있는 한도금액보다 초과하여 입력하면 Limit amount에 의해 다시 송금액 입력으로 돌아간다.</p> <p>(A)가 비밀번호를 잘못 입력하면 다시 비밀번호 입력으로 돌아가게 된다.</p>

Use Case	3. Deposit
Actor	User
Purpose	입금기능을 수행한다.
Overview	ATM기기를 통해 계좌에 입금한다.
Type	Primary and essential
Cross Reference	N/A
Pre-Requisites	해당 계좌의 정보가 txt파일에 있어야 한다
Typical Courses of Events	<p>(A): User (S): System</p> <p>1. (A)가 화면에 있는 빈칸에 계좌번호를 입력한다.</p> <p>2. (S)가 입력된 계좌번호에 해당되는 정보를 불러온다.</p> <p>3. (A)가 입금 버튼을 누른 다음 입금할 금액을 입력한다.</p> <p>4. (S)가 입금을 진행시키고 명세서를 뽑을지 말지 물어본다</p> <p>5. (A)가 Y/N을 눌러서 명세서를 뽑을지 말지 결정한다</p> <p>6. (S) 초기화면으로 되돌린다</p>
Alternative Course of Events	N/A
Exceptional Courses of Events	N/A

Use Case	4. Check Remain
Actor	User
Purpose	잔액조회기능을 수행한다.
Overview	ATM기기에서 잔액 조회를 한다
Type	Primary and essential
Cross Reference	<p>System function: 1.2</p> <p>Use Case: Check Password</p>
Pre-Requisites	해당 계좌의 정보가 txt파일에 있어야 한다

Typical Courses of Events	(A): User (S): System 1. (A)가 화면에 있는 빈칸에 계좌번호를 입력한다. 2. (S)가 입력된 계좌번호에 해당되는 정보를 불러온다 3. (A)가 잔액조회 버튼을 누른다 4. (S)가 비밀번호를 물어본다 5. (A)가 비밀번호를 입력한다 6. (S)가 비밀번호를 확인하고 잔액을 조회한 후 명세서를 뽑을지 말지 물어본다 7. (A)가 Y/N을 눌러서 명세서를 뽑을지 말지 결정한다 8. (S) 초기화면으로 되돌린다
Alternative Course of Events	N/A
Exceptional Courses of Events	N/A

Use Case	5. Print Statement
Actor	User
Purpose	영수증을 출력한다.
Overview	거래가 종료되고 나면 ATM기에서 명세서를 출력한다
Type	Primary and essential
Cross Reference	N/A
Pre-Requisites	거래가 완료된 상태여야 한다.
Typical Courses of Events	(A): User (S): System 1. (S)가 비밀번호를 확인하고 잔액을 조회한 후 명세서를 뽑을지 말지 물어본다 2. (A)가 Y/N을 눌러서 명세서를 뽑을지 말지 결정한다 3. (S) 초기화면으로 되돌린다
Alternative Course of Events	N/A
Exceptional Courses of Events	N/A

Use Case	6. Check Password
Actor	System
Purpose	User가 입력한 비밀번호가 맞는지 확인한다
Overview	user가 카드나 계좌번호를 입력한 후 비밀번호를 입력하면 이것을 해당 TXT파일에 있는 정보와 비교한다
Type	Primary and essential
Cross Reference	N/A
Pre-Requisites	해당 계좌에 대한 정보가 TXT파일에 있어야 한다.
Typical Courses of Events	(S): System

	1. (S)는 입력된 비밀번호를 확인하고 일치하면 다음 단계로 진행시킨다
Alternative Course of Events	N/A
Exceptional Courses of Events	비밀번호가 일치하지 않을 경우 시스템은 다시 비밀번호를 입력하는 화면을 출력한다.

Use Case	7. Limited Amount
Actor	System
Purpose	거래 액수가 정해 놓은 한도를 초과하는지 확인한다.
Overview	사용자가 출금/송금하려는 액수가 계좌에 정해 놓은 한도를 넘어서는지 확인한다.
Type	Primary and Essential
Cross Reference	System function: 2.1, 2.2 Use case: Send Money, Withdraw Money
Pre-Requisites	사용자가 금액을 입금하거나 송금하려는 상태여야 한다.
Typical Courses of Events	(S): System 1. (S-A) 입력받은 금액이 한도초과를 넘는지 검사한다 2. (S) 초과하지 않으면 거래를 진행시킨다
Alternative Courses of Events	N/A
Exceptional Courses of Events	한도를 넘어서게 되면 출금액이나 송금액을 다시 설정하도록 화면을 넘긴다.

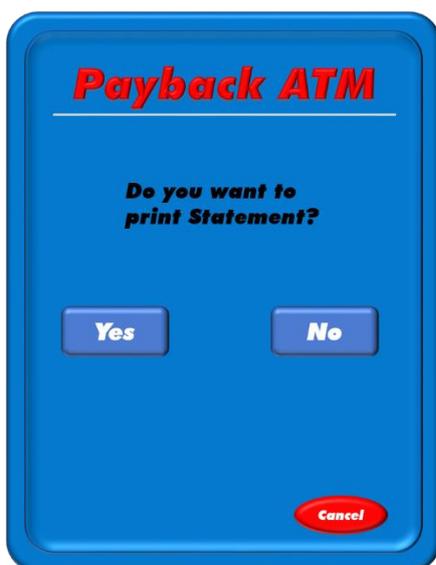
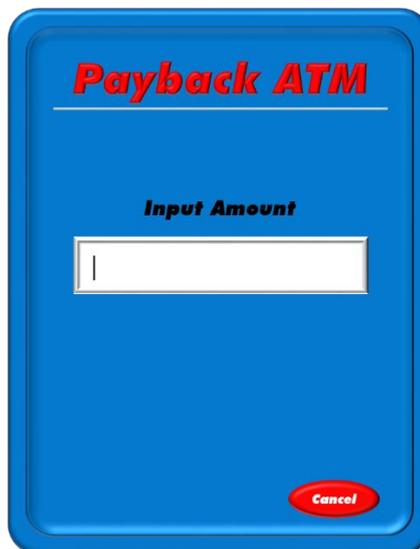
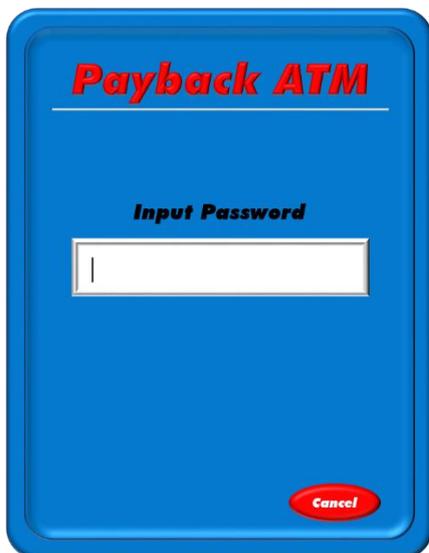
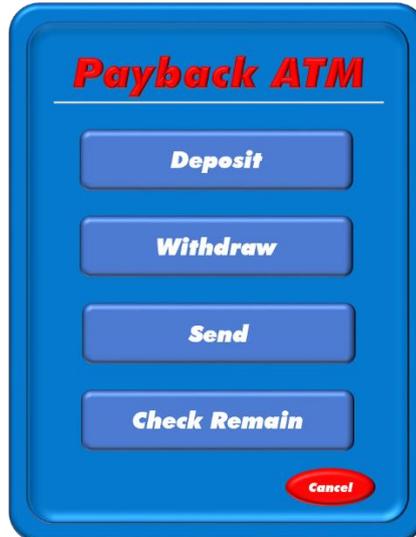
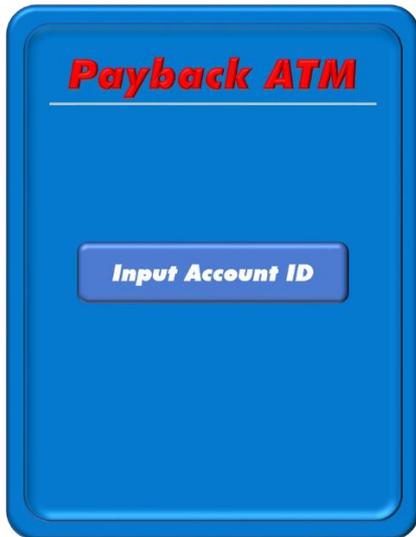
Use Case	8. Find Info
Actor	System
Purpose	거래를 하기 위해 시스템 내에 해당 계좌 정보를 적재한다.
Overview	입력한 계좌ID(통장, 카드, 계좌번호 등)에 맞는 계좌정보를 해당 은행에서 가져와 시스템 내에 적재한다.
Type	Primary and Essential
Cross Reference	N/A
Pre-Requisites	사용자가 계좌ID를 입력해야 한다.
Typical Courses of Events	(S): System 1. (S-A) 입력 받은 계좌번호의 정보를 가져온다.
Alternative Courses of Events	N/A
Exceptional Courses of Events	N/A

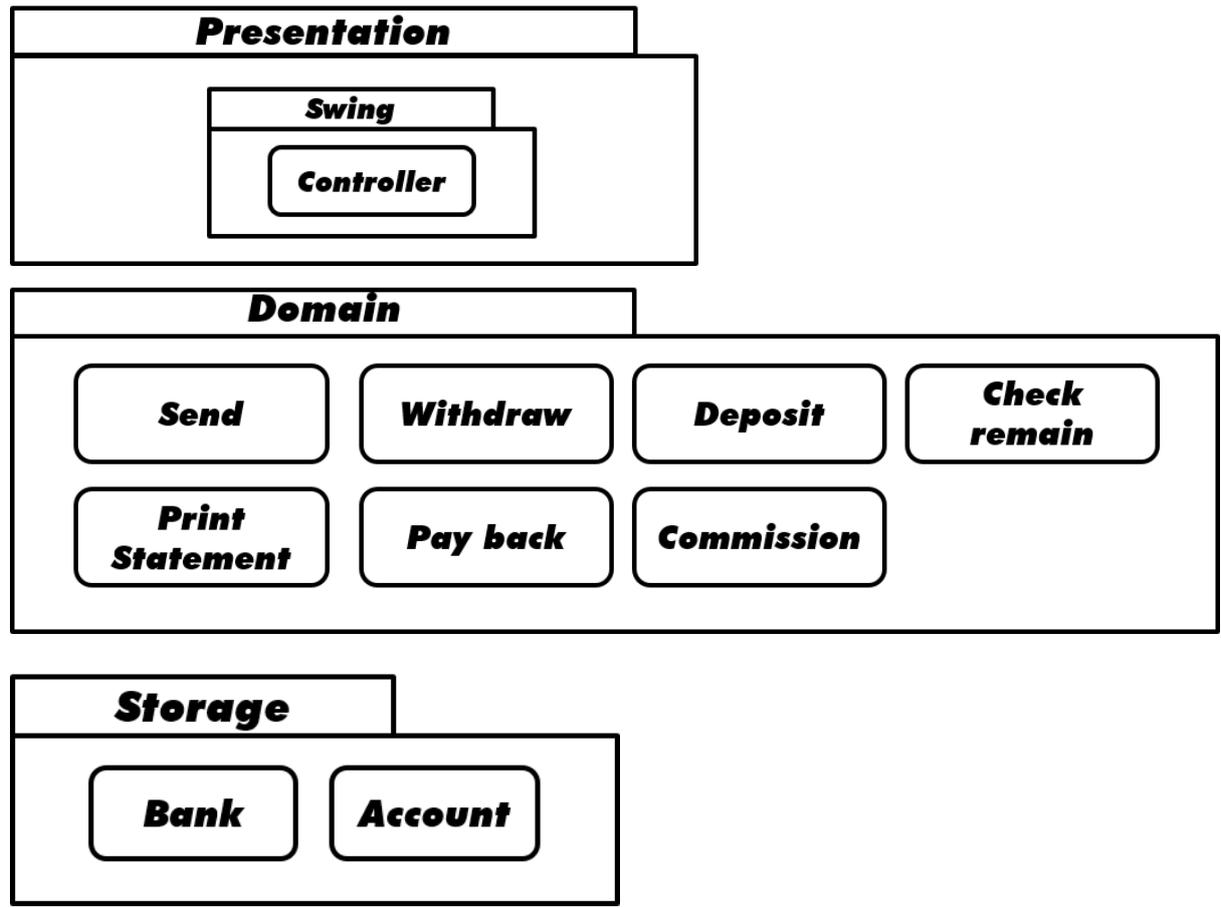
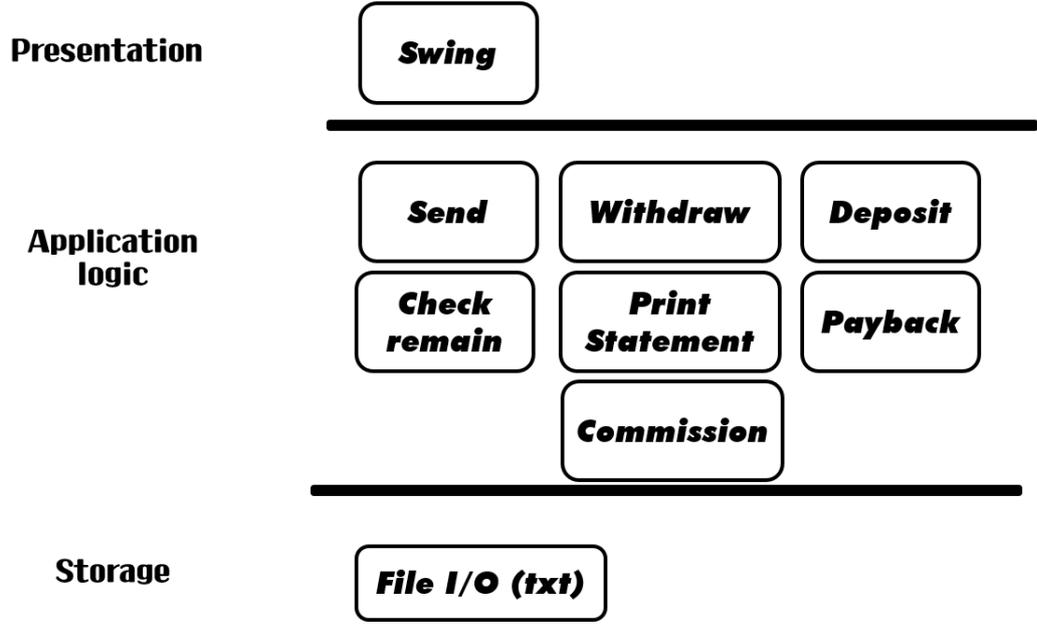
Use Case	9. Payback
----------	------------

Actor	System
Purpose	ATM 이용횟수에 따른 혜택을 제공한다.
Overview	모든 거래가 종료된 후 시스템 내에 적재된 계좌정보에 담겨있는 ATM 횟수를 조회하여 상당 이상이 되면 혜택을 제공한다.
Type	Primary and Essential
Cross Reference	N/A
Pre-Requisites	거래가 종료되어야 한다.
Typical Courses of Events	(S): System 1. (S)는 계좌를 갱신하는 동시에 계좌의 ATM 거래 횟수를 조회한다. 2. ATM 내에 정해 놓은 혜택 정책에 해당되면 (S)는 상당 상품의 기프트콘을 영수증을 통해 제공한다.
Alternative Courses of Events	N/A
Exceptional Courses of Events	해당 사항이 없으면 혜택 없이 종료한다.

Use Case	10. Count Commission
Actor	System
Purpose	거래에 필요한 수수료를 책정한다.
Overview	시스템에 Find Info로 적재한 계좌정보에서 은행의 영업시간을 조회하여 현재 시간과 비교하여 수수료를 책정한다.
Type	Primary and Essential
Cross Reference	System function: 2.1, 2.2 Use case: Send Money, Withdraw Money
Pre-Requisites	송금/ 출금 거래를 진행하고 있어야 한다.
Typical Courses of Events	(S): System 1. (S)는 Find Info로 적재한 계좌정보에서 은행의 영업시간을 조회한다. 2. (S)는 현재 시간과 비교하여 수수료를 책정한다. (영업시간 내 /외 판단, 각 은행별 수수료정책 조회)
Alternative Courses of Events	N/A
Exceptional Courses of Events	N/A

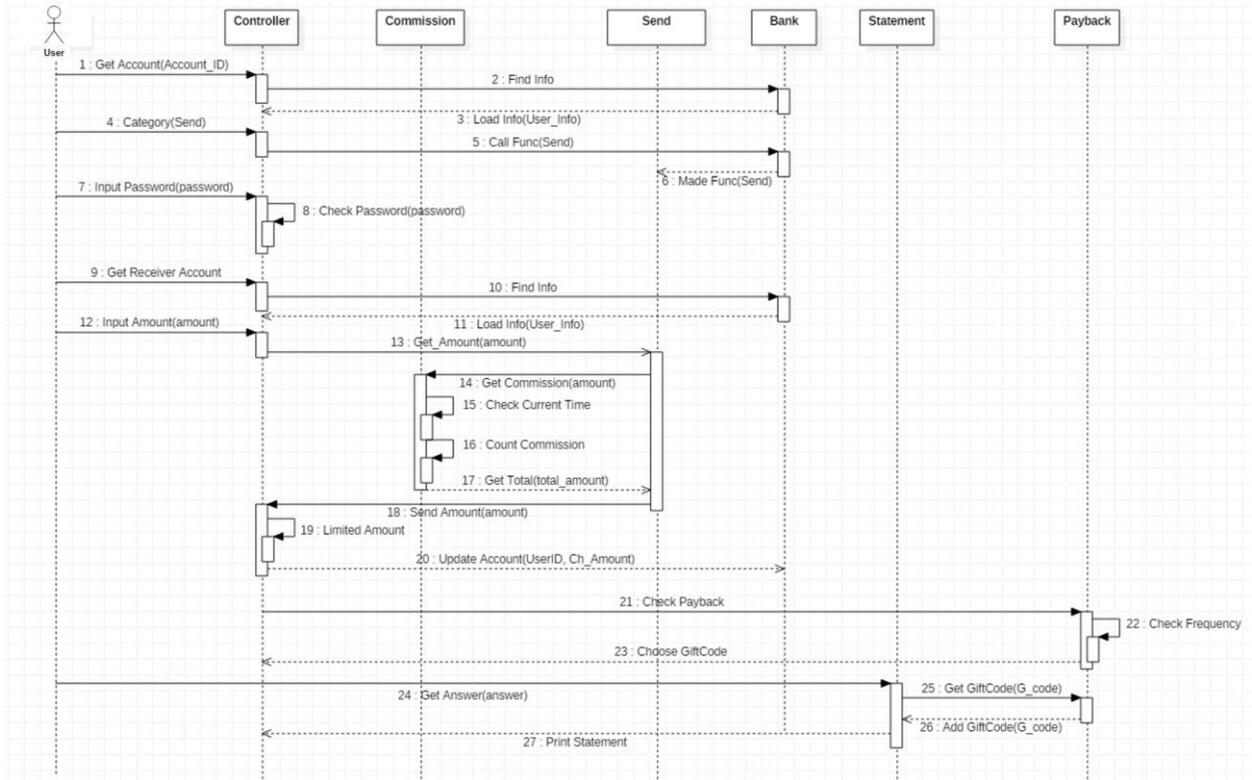
Activity 2042. Define Reports, UI, and Storyboards



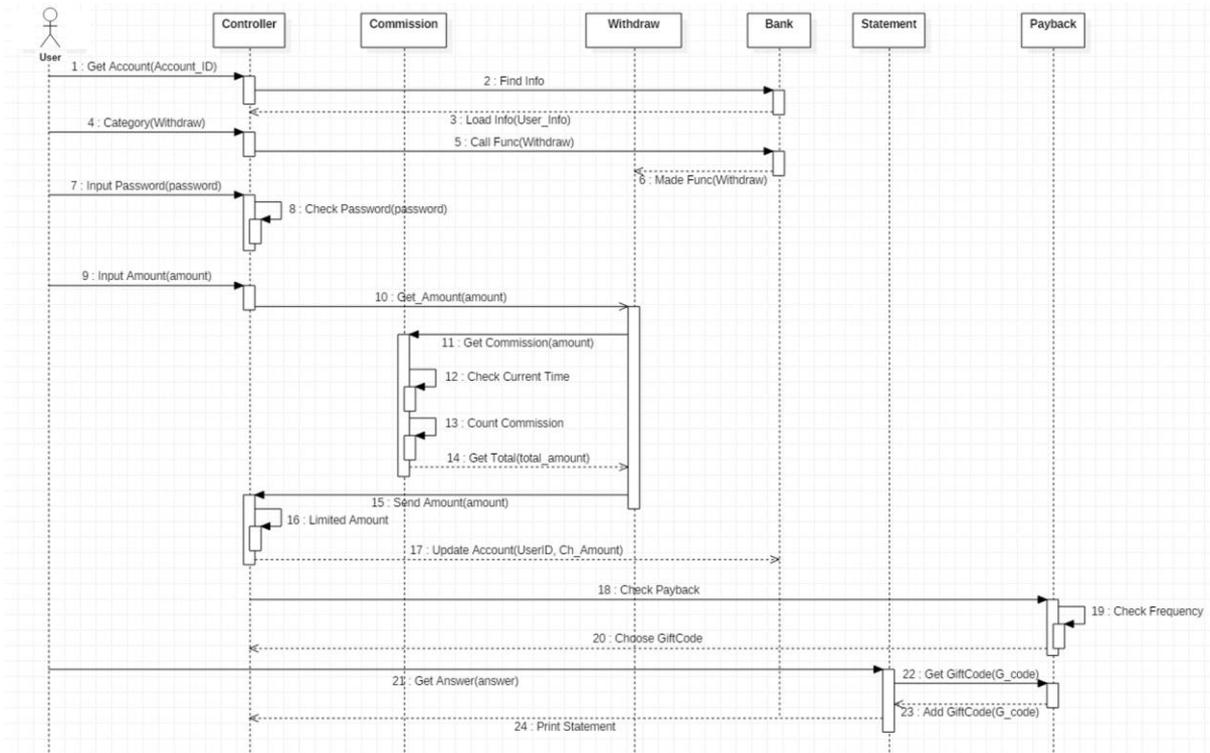


Activity 2044. Define Interaction Diagrams

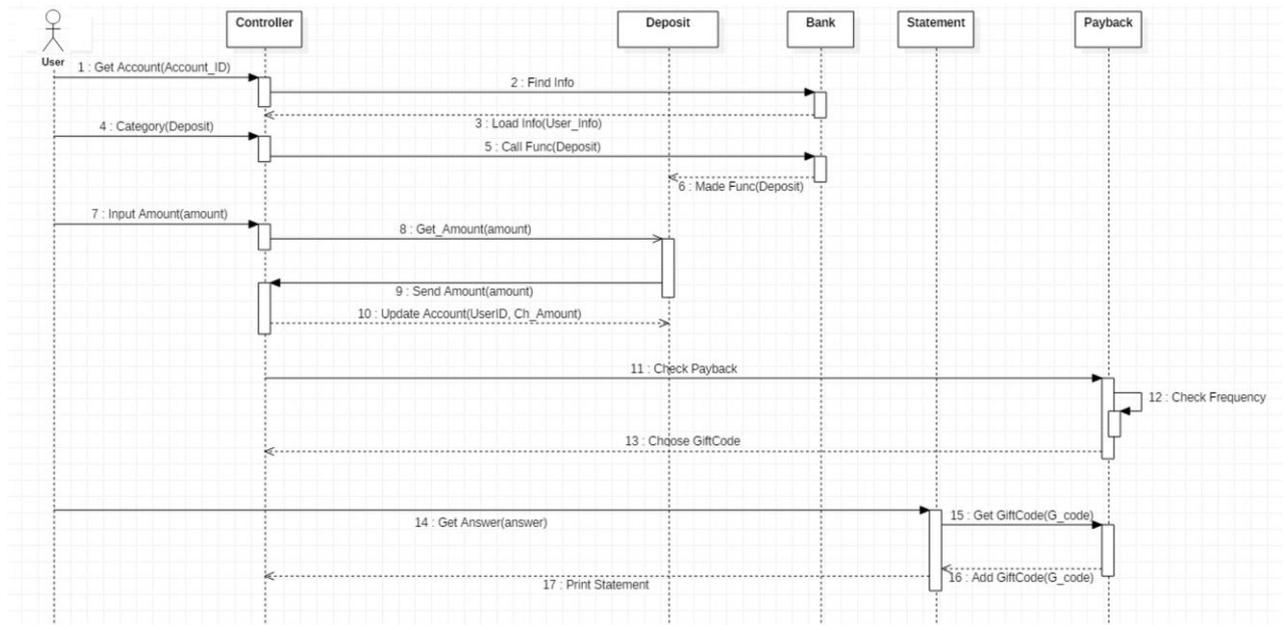
1) Send



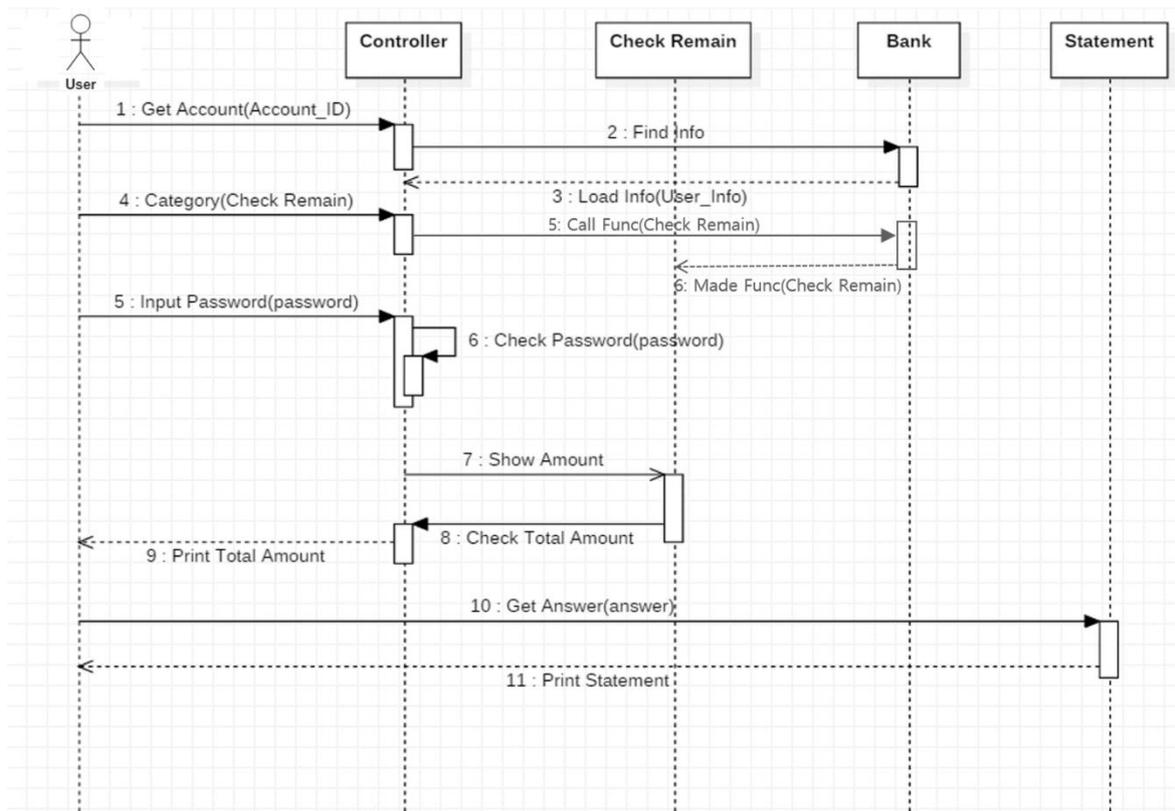
2) Withdraw



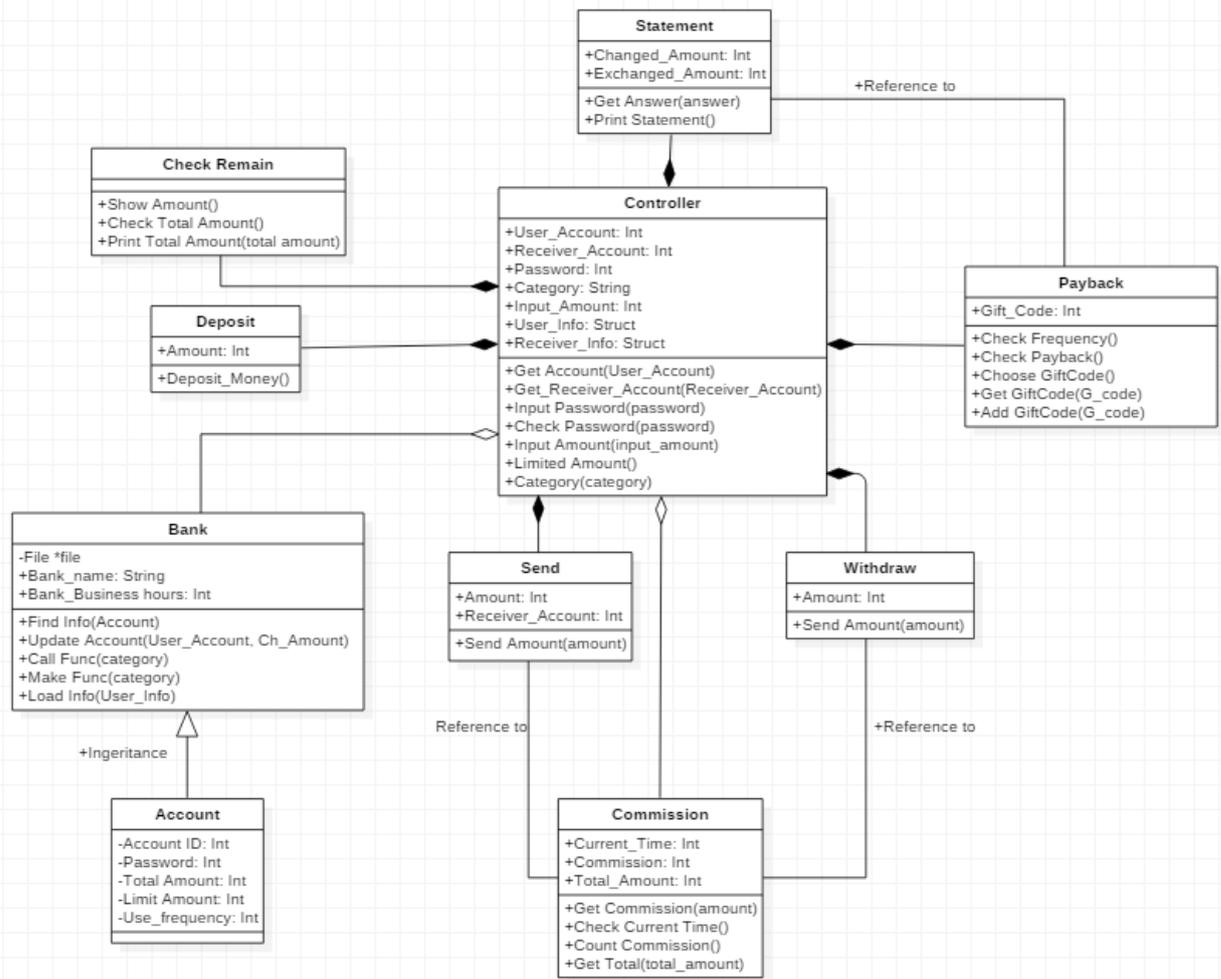
3) Deposit



4) Check remain



Activity 2045. Define Design Class Diagrams



Activity 2046. Design Traceability Analysis

Operation in Sequence Diagrams		Operation in Interaction Diagrams		Methods	Class
1. Get Account		Get Account(account_id)		Get Account(account_id: int): void	Controller
2. Category		Get Receiver_Account(account_id)		Get Receiver_Account(account_id: int): void	
2. Input Password		Input Password(password)		Input Password(password: int): void	
3. Get Receiver Account		Check Password(password)		Check Password(password: int): boolean	
4. Input Amount		Input Amount(amount)		Input Amount(amount: int): void	
5. Print Remain Amount		Check Limit(amount)		Check Limit(amount: int): Boolean	
6. Get Answer		Category(category)		Category(category: String): void	Bank
		Find Info(account_id)		Find Info(account_id : int): void	
		Load Info()		Load Info(): void	
		Call Func(category)		Call Func(category: String): void	
		Make Func(category)		Make Func(category: String): void	
		Update Account(User_id,Ch.amount)		Update Account(User_id: int,Ch.amount: int):void	
		Count Commission()		Count Commission(amount: int): int	Commission
		Get Commission(amount)		Get Commission(amount: int): void	
		Check Current Time()		Check Current Time(): int	
		Get Total(total_amount)		Get Total(total_amount)	Payback
		Check Payback(frequency)		Check Payback(frequency: int): void	
		Check Frequency(frequency)		Check Frequency(frequency: int): Boolean	
		Get Gift Code(G_code)		Get Gift Code(G_code: int): void	Payback
		Add Gift Code(G_code: int)		Add Gift Code(G_code: int): void	
		Choose Gift Code()		Choose Gift Code(): void	Statement
		Get Answer(answer)		Get Answer(answer: String): Boolean	
		Print Statement()		Print Statement(): void	Send/Withdraw
		Send Amount(amount)		Send Amount(amount: int): int	
		Show Amount()		Show Amount(): void	Check Remain
		Check Total Amount()		Check Total Amount(): int	
		Print Total Amount(total_amount)		Print Total Amount(total_amount: int): void	