

# Software Engineering 1

## Project: Instant Money Transfers and Bank-Accounts Access 24/7

### **Introduction:**

This document aimed at shedding light on the world of instant money transfers and bank account systems. In this document, we'll dive into the core concepts and advancements in these systems, aiming to provide a clear understanding of how they work and their evolution over time. Additionally, we'll explore the stakeholders involved in these systems and their interests, as well as outline the essential functions these systems perform. By breaking down these complex topics into simple terms, we hope to equip readers with the knowledge needed to navigate the realm of modern finance confidently.

### **Scope:**

The application of money transfer and bank account systems represents an intermediary platform that facilitates seamless interactions between users and banking institutions. This system enables users to initiate money transfers to other users' bank accounts or digital wallets, settle various bills and donations, request funds from other users, and split bills across multiple accounts. Additionally, it offers the convenience of scheduling bill payments, enhancing efficiency in financial management for users.

<b>Word</b>	<b>Definition</b>
User	Who's want to access bank card to make transferring money with easily access
Admin	Person who manages system as add bank account and show statistics through error tracking
Database	Collection of all the information monitored by this system.
Software Requirements Specification	A document that completely describes all of the functions of a proposed system and the constraints under which it must operate. For example, this document.
Stakeholder	Any person with an interest for success project
UML Diagram	Is visual modeling language that is intended to provide a standard way to visualize the design of a system.
FAQ	frequently asked questions is question in a list of questions and answers intended to help people understand a particular subject

# Table of Contents

Introduction:.....	1
Scope:.....	1
System Requirement .....	5
Function Requirement: .....	5
Non-Function Requirement: .....	8
UML Diagram.....	10
1. Use Case Diagram: .....	10
1.1 Use Case Description: .....	11
2. Activity Diagram: .....	34
2.1 Login & Register.....	34
2.2 Send Money.....	35
2.3 Receive Money.....	36
2.4 Card Modify .....	37
2.5 Check Balance.....	38
2.6 View Transaction History .....	39
2.7 Reset Password.....	40
2.8 Admin Add Bank and modify .....	41
2.9 Customer Service and help.....	42
3. Sequence Diagram: .....	43
3.1 Login SSD .....	43
3.2 Send Money SSD .....	44
3.3 Request Money SSD .....	45
3.4 Add Card SSD .....	46
3.5 User Delete Account .....	47
3.6 Reset Password SSD .....	48
3.7 Admin Add Bank SSD .....	49
3.8 Admin View User Account.....	50

4. System Architecture: .....	51
5. Class Diagram: .....	52
6. ERD Diagram: .....	53
7. Package Diagram: .....	55
8. Object Diagram: .....	57
9. Snapshots: .....	58

# System Requirement

## Function Requirement:

- 1- The system allows users to make registration and create account with name, email, phone number, gender, password, photo (to be able to login next).
- 2- The system allows users to login with email and password that entered in registration form before.
- 3- The system allows users to login with face recognition and password that entered in registration form before.
- 4- The system must allow users to add card to system by enter information (card name holder, card number, verification 3 digits, bank name).
- 5- The system must allow user to enter card number by scanner OCR.
- 6- The system must allow users to edit card details (Change Card name).
- 7- The system allows user to drop bank card.
- 8- The system must enable users to transfer money to another user through enter card number for the other one.

- 9- User can transfer money to another one through his/her wallet instead of bank number.
- 10- User can send money for donation.
- 11- User can pay payment through system.
- 12- The system must allow users to confirm checkout at last step in transfer money.
- 13- The system must allow users to manage and edit account info (account name, email, phone number).
- 14- The system must allow users to reset account password.
- 15- The system must allow users to delete account.
- 16- The system must enable users to export report mini statement include (balance and view last transaction on their account).
- 17- The system must allow users to view transaction history.
- 18- The system must allow users to check balance for the card.
- 19- The system must allow users to receive money by send request to other user with amount of money need.
- 20- The system must allow users to confirm requests that other users sent and transfer money.
- 21- The systems must enable users to view transaction status.
- 22- The system allows users to view exchange rates of money.
- 23- The system must enable users to make list of favorite cards.

- 24- The system allows users to make bill schedule which can set date automatic for bill withdraw.
- 25- The system allows users to split bill for each card and determine the amount of money assigned to other card and date that money taken.
- 26- The system must enable users to make FAQ with bank to answer question.
- 27- The system must allow users to send feedback.
- 28- The system shall provide help to guide the user.
- 29- The admin shall add bank / wallet.
- 30- The admin shall add donation account
- 31- The admin shall add bill payment account.
- 32- The admin shall view error tracking analysis.
- 33- The admin shall view users of system.
- 34- The system shall verify user account during login.
- 35- The system shall verify card number with bank before add card.
- 36- The system shall check IPN to users before transferring done.

## **Non-Function Requirement:**

### **1. Performance:**

- The system shall transfer money progress for user in less than 5 second.
- The system shall respond to the request from the user in less than 3 seconds.
- The application should ideally function continuously, aiming to minimize instances of failure and maximize uptime.

### **2. Maintainability:**

- The system shall easy to be able change and add feature in the future.

### **3. Security:**

- The system must encrypt user password and card number and shall protect user data.
- The system shall distinguish between authorized and non- authorized users.



#### **4. Manageability:**

- The system must be designed for efficiency, with simple monitoring and maintenance processes in place to ensure consistent performance, security, and smooth operation.

#### **5. Interface:**

- The system must be ease to use and user for first time can use functionality.

#### **6. Look and feel:**

- The system must be comfortable to eyes and icons appear without any obstacles.

#### **7. Availability:**

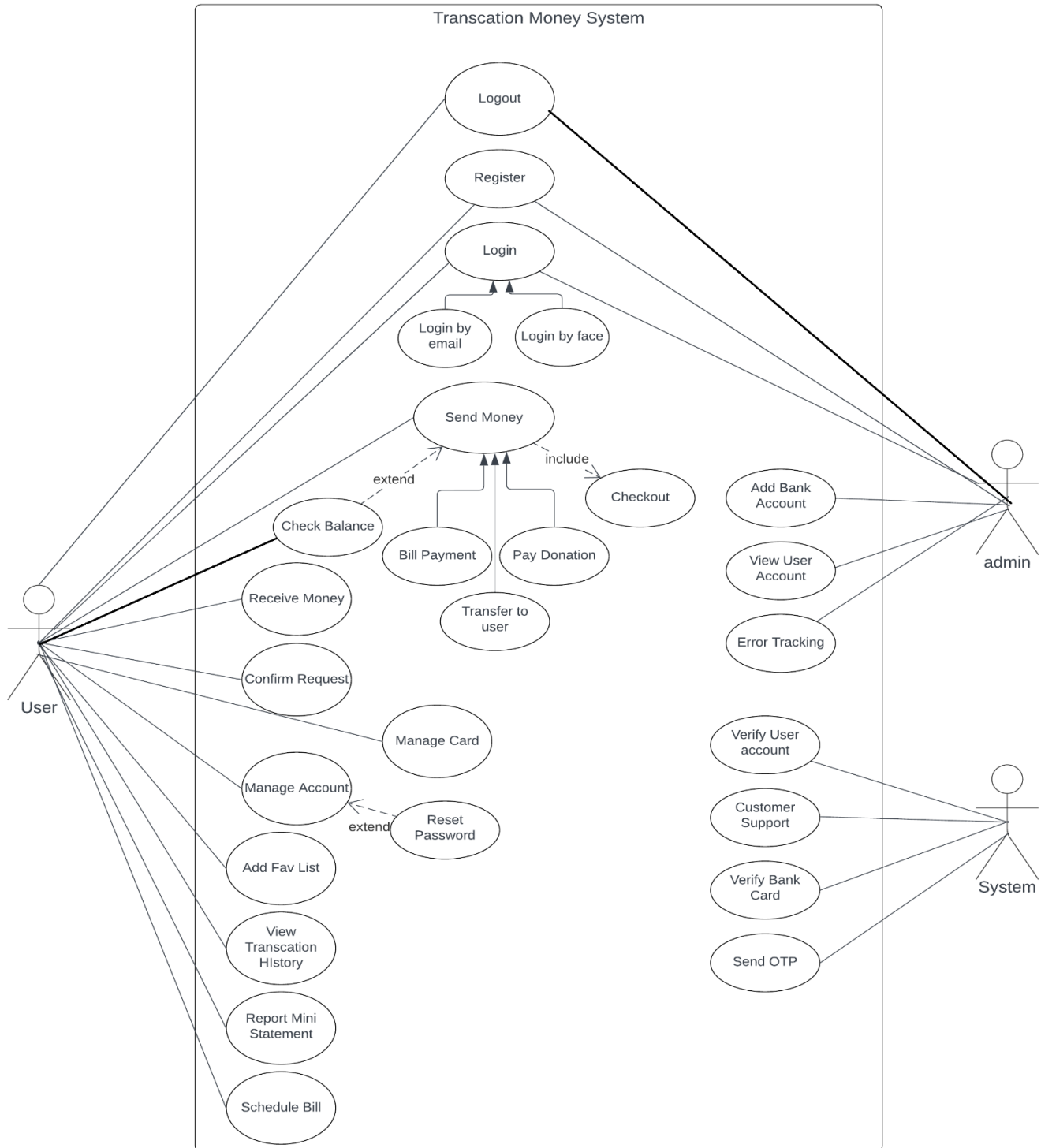
- The system shall be available 24 hours in a day.

#### **8. Reliability:**

- The system shall ensure that transactions are processed accurately and securely within a maximum time frame of 10 seconds, with a failure rate not exceeding 0.1% of total transactions per month.

# UML Diagram

## 1. Use Case Diagram:



## 1.1 Use Case Description:

Registration Description	
ID Number -Name	UC-1 Register
Goal	User enter his/her data to create account in system to be able to use system function
Actor(s)	User (Person)- Admin
Preconditions	----
Postconditions	Account is added and registration successfully
Error Situations	User not enter data in form field so cannot register
Main Success Scenario	<ol style="list-style-type: none"><li>1. Open register page</li><li>2. Enter his/her data</li><li>3. Submit data</li><li>4. System Confirm that registration is done</li></ol>
Alternative Processes	<ol style="list-style-type: none"><li>2'. User enter invalid data</li><li>3'. Show error message until correct then submit</li></ol>

Login Description	
ID Number -Name	UC-2 Login
Goal	User login to system to use function and transfer money
Actor(s)	User (Person) - Admin
Preconditions	Register before
Postconditions	Login successfully and turn to home page
Error Situations	User not enter data in form field so cannot login
Main Success Scenario	<ol style="list-style-type: none"> <li>1. Open login page</li> <li>2. Enter email/Face and password</li> <li>3. System check account</li> <li>4. Turn to home page</li> </ol>
Alternative Processes	<ol style="list-style-type: none"> <li>3'. User not have account</li> <li>4'. Turn to register page</li> </ol>

Send Money Description	
ID Number -Name	UC-3 Send Money
Goal	User transfer money to another user
Actor(s)	User (Person)
Preconditions	Login before
Postconditions	The money transferred and balance minus the amount
Error Situations	User do not have enough balance to transfer
Main Success Scenario	<ol style="list-style-type: none"> <li>1. Open transfer page</li> <li>2. Enter card number and amount of money</li> <li>3. Confirm the data entered</li> <li>4. Transfer money</li> <li>5. balance changed</li> </ol>
Alternative Processes	<ol style="list-style-type: none"> <li>2'. Enter card number incorrect</li> <li>3'. Error message appear with problem then confirm again</li> </ol>

Bill payment Description	
ID Number -Name	UC-4 Bill payment
Goal	User send money to pay payment
Actor(s)	User (Person)
Preconditions	Login before
Postconditions	The money sent and balance minus the amount
Error Situations	User do not have enough balance to transfer
Main Success Scenario	<ol style="list-style-type: none"> <li>1. Open bill payment page</li> <li>2. Choose type of bill need to pay</li> <li>3. Enter amount of money</li> <li>4. Confirm the data entered</li> <li>5. send money</li> <li>6. balance changed</li> </ol>
Alternative Processes	<ol style="list-style-type: none"> <li>5'. Balance not enough</li> <li>6'. Error message appear and with until recharge and make the process again</li> </ol>

Donation Description	
ID Number -Name	UC-5 Donation
Goal	User send money for Donation
Actor(s)	User (Person)
Preconditions	Login before
Postconditions	The money sent and balance minus the amount
Error Situations	User do not have enough balance to transfer
Main Success Scenario	<ol style="list-style-type: none"> <li>1. Open Donation page</li> <li>2. Choose Donation type</li> <li>3. Enter amount of money</li> <li>4. Confirm the data entered</li> <li>5. Send money</li> <li>6. balance changed</li> </ol>
Alternative Processes	<ol style="list-style-type: none"> <li>5'. Balance not enough</li> <li>6'. Error message appear and with until recharge and make the process again</li> </ol>

Check Balance Description	
ID Number -Name	UC-6 Check Balance
Goal	User want to check amount of money inside the Card
Actor(s)	User (Person)
Preconditions	Login before
Postconditions	The user knows his/her balance
Error Situations	Not login
Main Success Scenario	<ol style="list-style-type: none"> <li>1. Press icon check balance</li> <li>2. Enter secure IPN</li> <li>3. Balance appears</li> </ol>
Alternative Processes	<ol style="list-style-type: none"> <li>2'. Invalid IPN Entered</li> <li>3'. Error message appear and enter Correct IPN again</li> <li>4'. Balance appears</li> </ol>



Checkout Description	
ID Number -Name	UC-7 Checkout
Goal	User checkout last step in transfer that all data is correct
Actor(s)	User (Person)
Preconditions	Login before and enter all details about transferring
Postconditions	Transfer Process Success and balance change
Error Situations	-----
Main Success Scenario	<ol style="list-style-type: none"> <li>1. Check all details entered</li> <li>2. Confirm Checkout</li> </ol>

Receive Money Description	
ID Number -Name	UC-8 Receive Money
Goal	User want to receive money so send request
Actor(s)	User (Person)
Preconditions	Login before
Postconditions	Send request money to user want to send money
Error Situations	The entered other user Card number invalid
Main Success Scenario	<ol style="list-style-type: none"> <li>1. Enter Card Number and Name of another user</li> <li>2. Enter amount of money need</li> <li>3. Send Request</li> </ol>

Confirm Request Description	
<b>ID Number -Name</b>	UC-9 Confirm Request
<b>Goal</b>	User Confirm requests of money and send to users
<b>Actor(s)</b>	User (Person)
<b>Preconditions</b>	Login before
<b>Postconditions</b>	Send money to person that request money
<b>Error Situations</b>	Not have enough money to send
<b>Main Success Scenario</b>	<ol style="list-style-type: none"> <li>1. Open Requests list page</li> <li>2. Select which request need to send</li> <li>3. Confirm Request</li> </ol>
<b>Alternative Processes</b>	<ol style="list-style-type: none"> <li>3'. Balance not enough</li> <li>4'. Error message appear and recharge money then send</li> <li>5''. Decline the request</li> </ol>

Manage Card (Add Card) Description	
ID Number -Name	UC-ID Manage Card (Add Card)
Goal	User can add card
Actor(s)	User (Person)
Preconditions	Login before
Postconditions	User add card in his/her account
Error Situations	Card number not correct
Main Success Scenario	<ol style="list-style-type: none"> <li>1. Open manage card page</li> <li>2. Choose add card</li> <li>3. Enter card details (Name-card number-phone number-bank)</li> <li>4. Confirm card details</li> </ol>
Alternative Processes	<ol style="list-style-type: none"> <li>4'. Card number not involved in bank database</li> <li>5'. Contact to bank or Call Customer Services</li> </ol>

Manage Card (drop Card) Description	
ID Number -Name	UC-II Manage Card (drop Card)
Goal	User need to delete card
Actor(s)	User (Person)
Preconditions	Login before
Postconditions	User drop card in his/her account
Error Situations	---
Main Success Scenario	<ol style="list-style-type: none"> <li>1. Open manage card page</li> <li>2. Choose delete card</li> </ol>

Reset Password Description	
ID Number -Name	UC-12 Reset Password
Goal	User need to change account password
Actor(s)	User (Person)
Preconditions	Login before
Postconditions	User account password change
Error Situations	Enter not matched password
Main Success Scenario	<ol style="list-style-type: none"> <li>1. Open manage account page</li> <li>2. Choose reset password</li> <li>3. Enter new password</li> <li>4. Enter confirmed password again</li> </ol>
Alternative Processes	<ol style="list-style-type: none"> <li>4'. Password not matched</li> <li>5'. Error message appear and enter again</li> </ol>

Add Favorite list Description	
ID Number -Name	UC-13 Add Favorite list
Goal	User save card number favorite
Actor(s)	User (Person)
Preconditions	Login before
Postconditions	Entered Favorite card number
Error Situations	Card number not found
Main Success Scenario	<ol style="list-style-type: none"> <li>1. Open favorite card page</li> <li>2. Enter card number</li> <li>3. Checked valid number</li> <li>4. Confirm added card to favorite</li> </ol>
Alternative Processes	<ol style="list-style-type: none"> <li>3'. Card number not found</li> <li>4'. Renter card number</li> <li>5'. Check valid number</li> </ol>

View Transaction History Description	
ID Number -Name	UC-14 View Transaction History
Goal	Enable user to see last Transactions
Actor(s)	User (Person)
Preconditions	Login before and make transaction
Postconditions	User see list of transaction
Error Situations	-----
Main Success Scenario	<ol style="list-style-type: none"> <li>1. Open Transaction History Page</li> <li>2. Enter IPN Number</li> <li>3. Confirm IPN and view list of history</li> </ol>
Alternative Processes	<ol style="list-style-type: none"> <li>3'. IPN not correct</li> <li>4'. Error message appear and reenter again</li> <li>5'. View List of history</li> </ol>



Report Mini Statement Description	
ID Number -Name	UC-15 Report Mini Statement
Goal	User need to export file include balance and last transaction
Actor(s)	User (Person)
Preconditions	Login before and make transaction
Postconditions	File Exported with information
Main Success Scenario	<ol style="list-style-type: none"> <li>1. Open Report Page</li> <li>2. Export Mini Statement</li> </ol>

Schedule Bill Description	
ID Number -Name	UC-16 Schedule Bill
Goal	User can schedule bill automatic by assign date of transfer automatic monthly
Actor(s)	User (Person)
Preconditions	Login before and enter bill type
Postconditions	Assigned date for automatic transfer
Error Situations	Bill payment date change randomly
Main Success Scenario	<ol style="list-style-type: none"> <li>1. Open bill Schedule page</li> <li>2. Select type of bill need</li> <li>3. Select date</li> <li>4. Confirm Bill schedule</li> </ol>

Add Bank Account Description	
ID Number -Name	UC-17 Add Bank Account
Goal	Admin must able to enter bank account
Actor(s)	Admin
Preconditions	Login before and enter bill type
Postconditions	Bank added to list of banks and user can add his/her card related to bank
Error Situations	Bank information and connection no correct
Main Success Scenario	<ol style="list-style-type: none"> <li>1. Open add account page</li> <li>2. Select add bank account</li> <li>3. Enter bank information</li> <li>4. Check bank information with bank and approve</li> <li>5. Bank added to list of banks</li> </ol>
Alternative Processes	<ol style="list-style-type: none"> <li>3'. Enter invalid bank information</li> <li>4'. Renter details and information again</li> <li>5'. Verify with bank about correct information</li> <li>6'. Bank added to list of banks</li> </ol>

View User Account Description	
ID Number -Name	UC-18 View User Account
Goal	Admin can see all user registered
Actor(s)	Admin
Preconditions	Login before
Postconditions	Admin see list of user accounts
Error Situations	Cannot receive user accounts added
Main Success Scenario	1. Choose view user account list

Error Tracking Description	
ID Number -Name	UC-19 Error Tracking
Goal	Admin see error analysis happen in system in transfer process of user (Failed Transaction)
Actor(s)	Admin
Preconditions	Login before
Postconditions	Admin see error analysis
Main Success Scenario	1. Choose Error Tracking page

Verify User Account Description	
ID Number -Name	UC-20 Verify User Account
Goal	System must verify account of user in login phase before login
Actor(s)	System
Preconditions	----
Postconditions	System approve that this user is exists or not and open home page and use function if exist if not turn to register page
Main Success Scenario	<ol style="list-style-type: none"> <li>1. User Enter email and password</li> <li>2. System check that account exists</li> <li>3. Login success and turn to home page</li> </ol>

Verify Card Number Description	
ID Number -Name	UC-21 Verify Card Number
Goal	System must verify user card Number through add card with bank
Actor(s)	System
Preconditions	User have card number and account in real bank Database
Postconditions	System approve that this user card is exists or not in bank
Main Success Scenario	<ol style="list-style-type: none"> <li>1. User Enter card number and phone number</li> <li>2. System check that card exists</li> <li>3. Confirm that card and added to list</li> </ol>

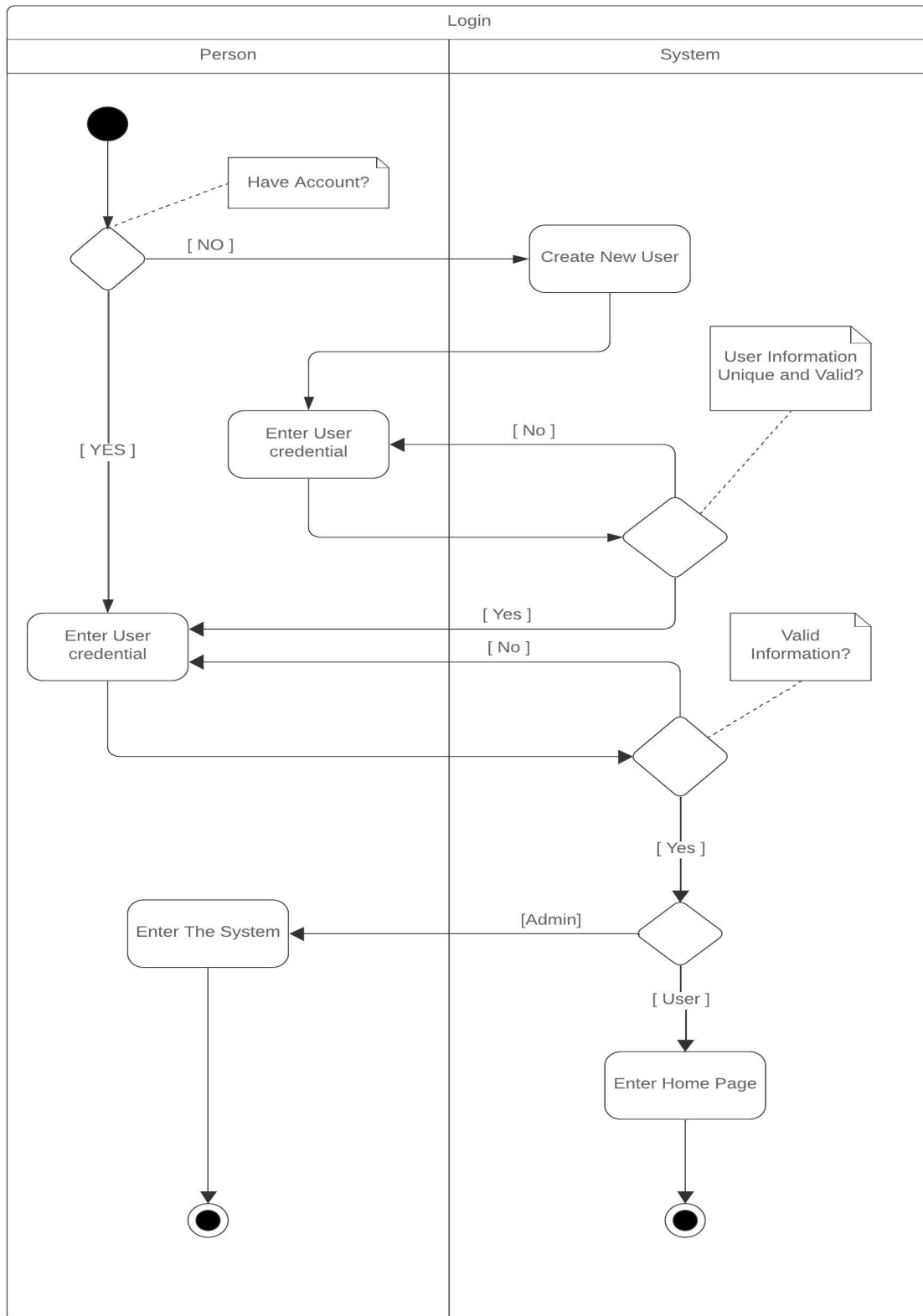
Send OTP Description	
<b>ID Number -Name</b>	UC-22 Send OTP
<b>Goal</b>	System must send OTP to user in registration to check and verify email enter
<b>Actor(s)</b>	System
<b>Preconditions</b>	---
<b>Postconditions</b>	User get OTP and enter it to confirm registration
<b>Main Success Scenario</b>	<ol style="list-style-type: none"> <li>1. User Enter all user data</li> <li>2. System send OTP</li> <li>3. User enter OTP and system match it</li> <li>4. Registration Done</li> </ol>
<b>Alternative Processes</b>	<ol style="list-style-type: none"> <li>3'. OTP Entered not Correct</li> <li>4'. Resend request</li> <li>5'. System resend OTP</li> <li>6'. User enter OTP and system match it</li> <li>7'. Registration Done</li> </ol>



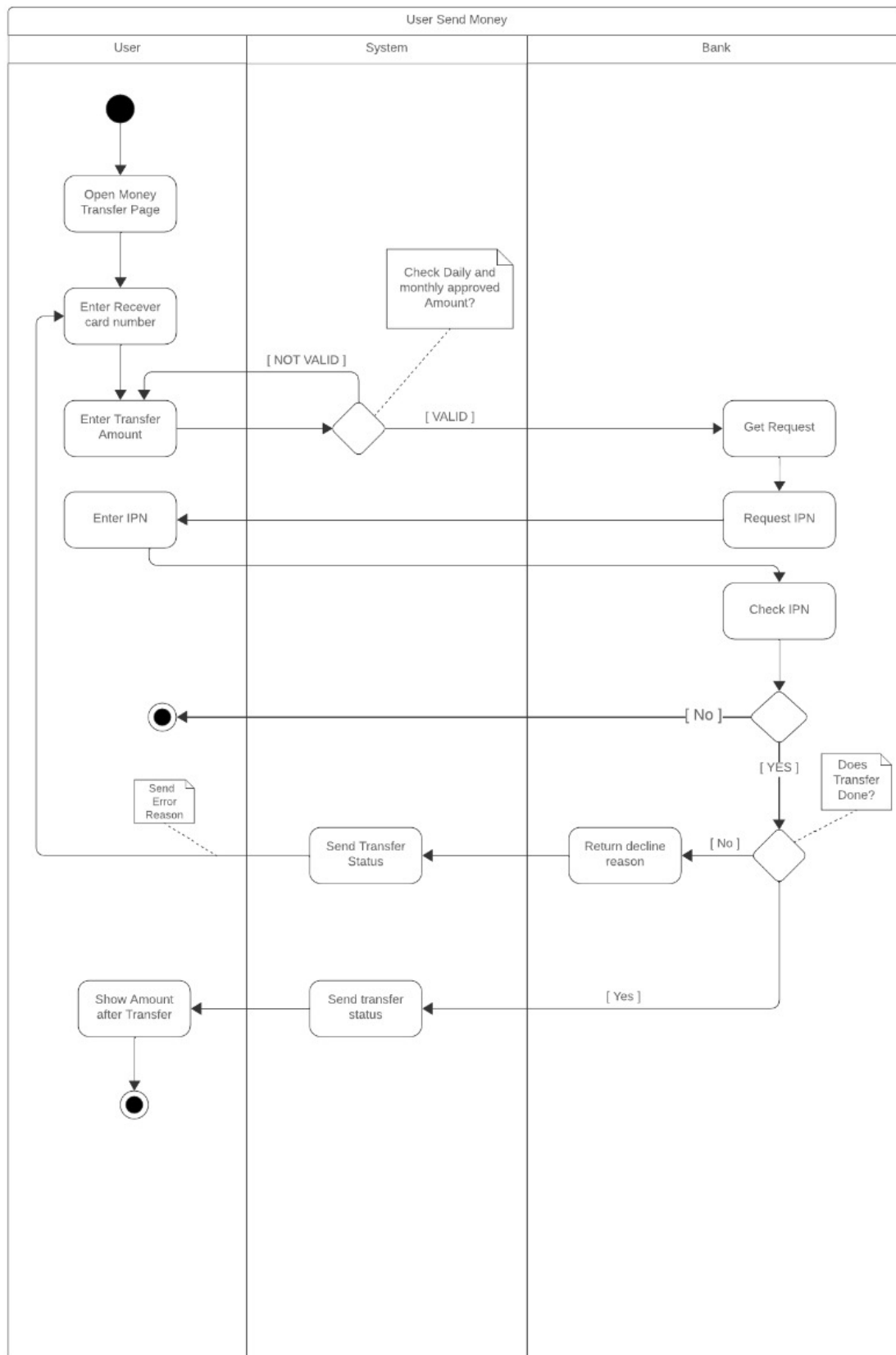
Customer Support Description	
ID Number -Name	UC-23 Customer Support
Goal	System must manage any problem face user and user send question or problem to system
Actor(s)	System
Preconditions	User need to ask question
Postconditions	User get answer and solve his/her problem
Main Success Scenario	<ol style="list-style-type: none"> <li>1. User send question</li> <li>2. System get request</li> <li>3. Customer Service send answer to user</li> <li>4. User can solve his/her problem</li> </ol>

## 2. Activity Diagram:

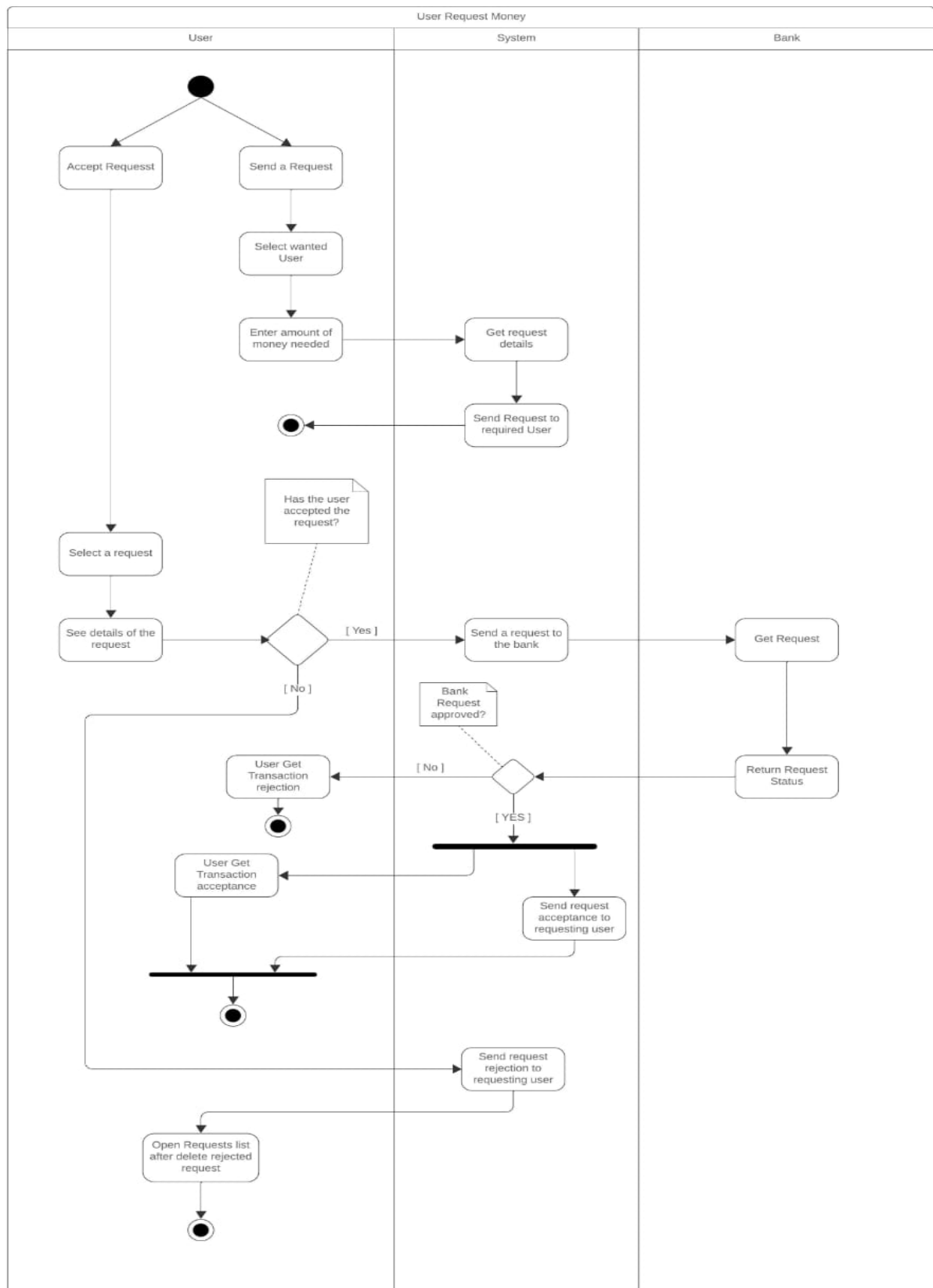
### 2.1 Login & Register



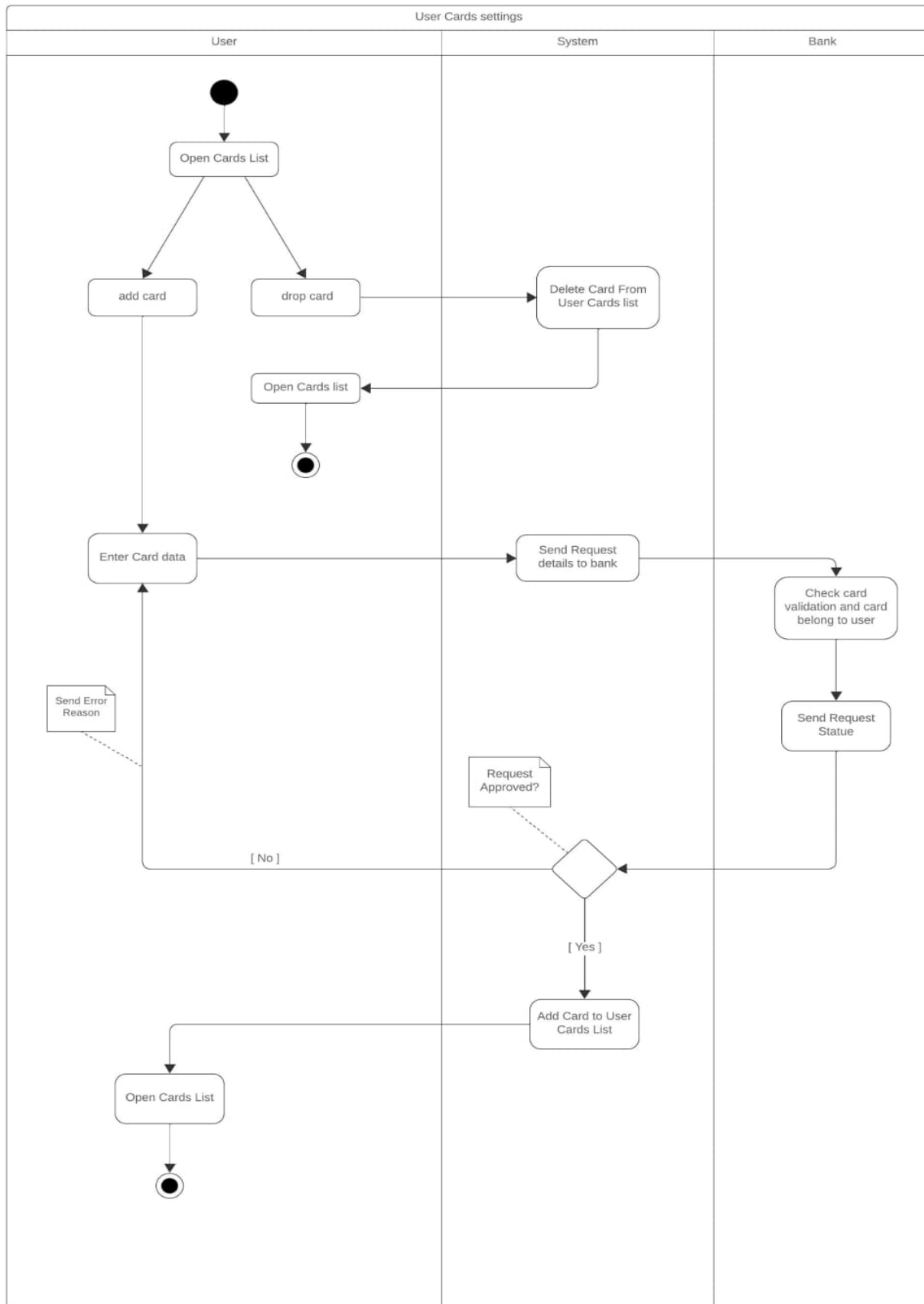
## 2.2 Send Money



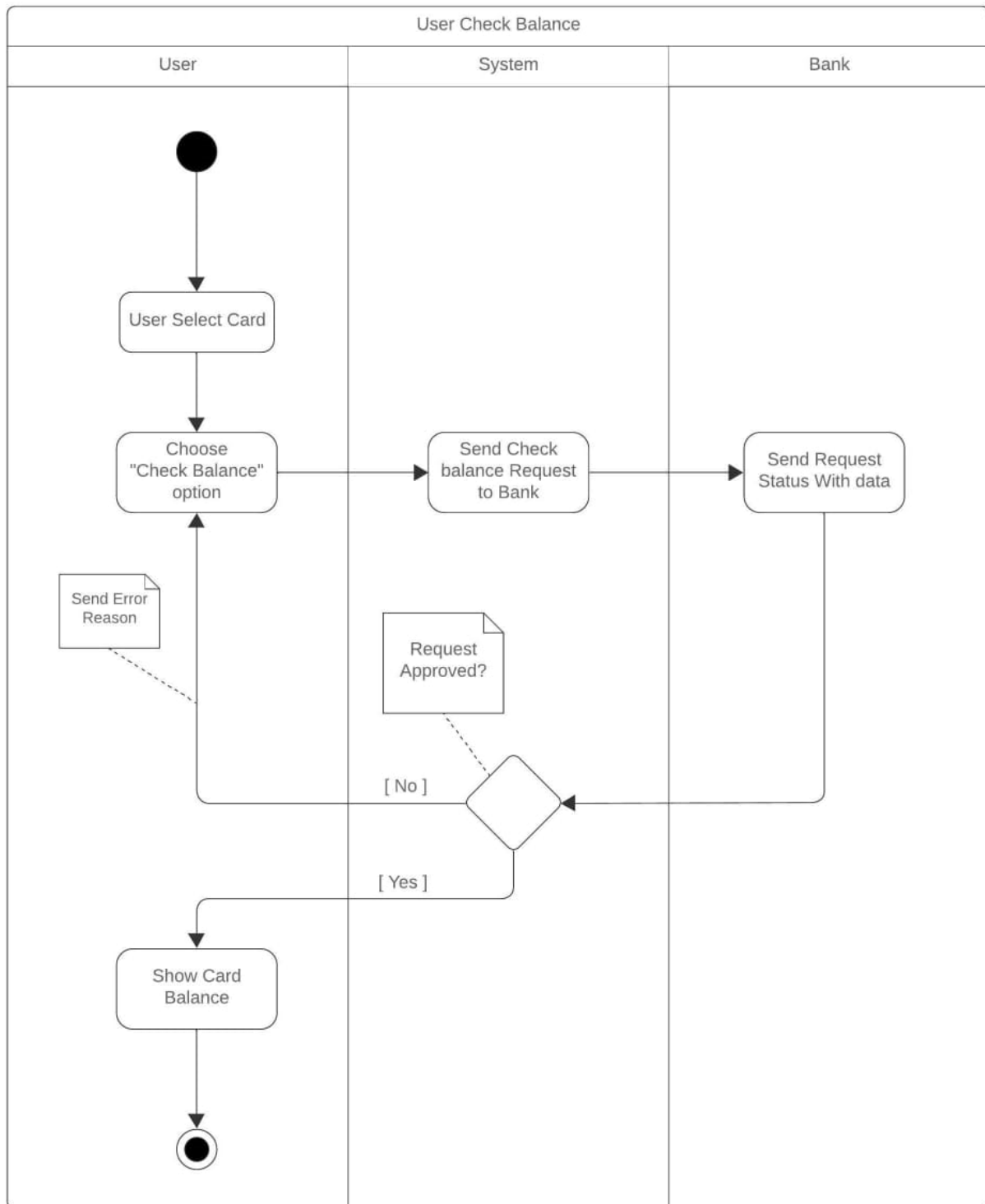
## 2.3 Receive Money



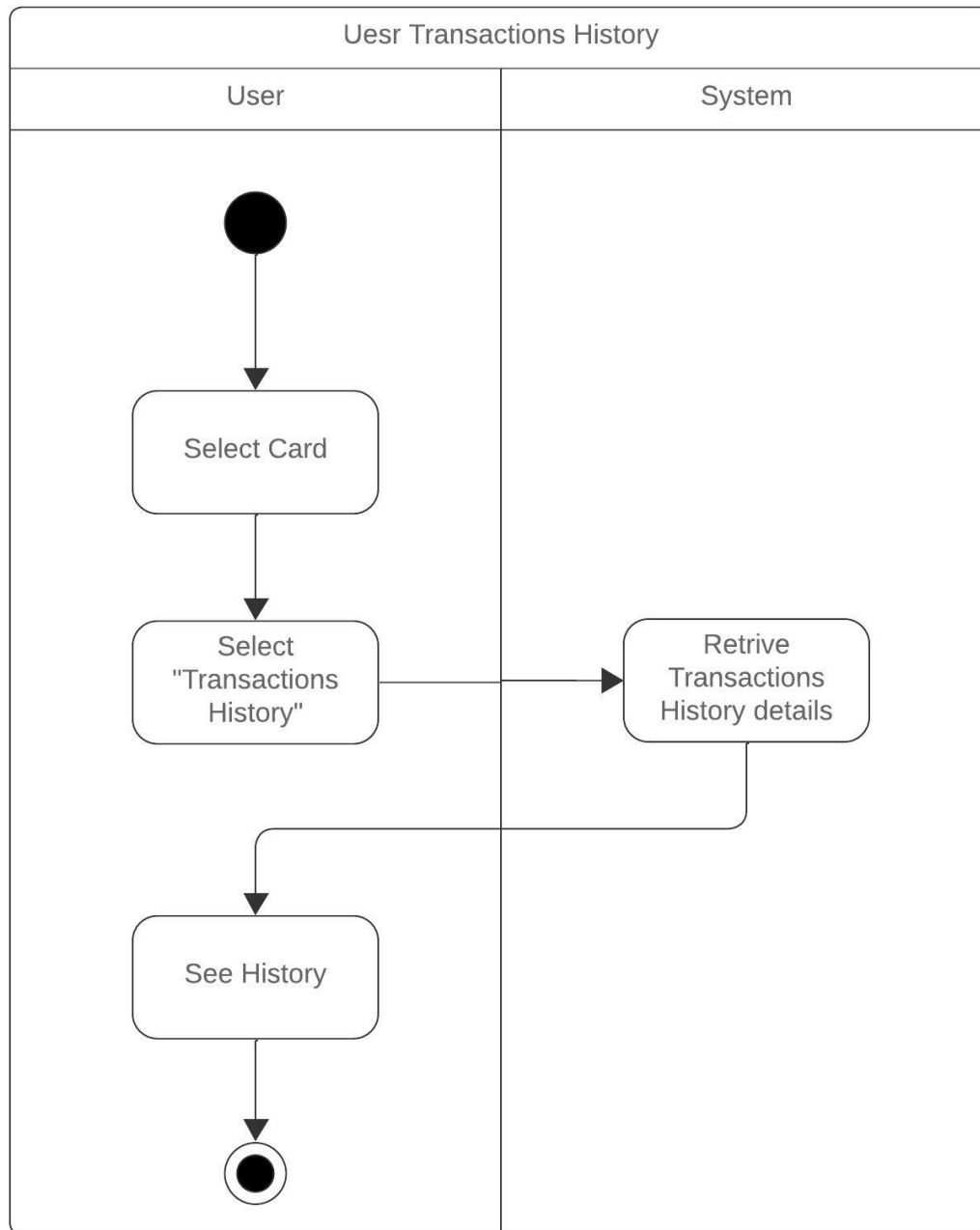
## 2.4 Card Modify



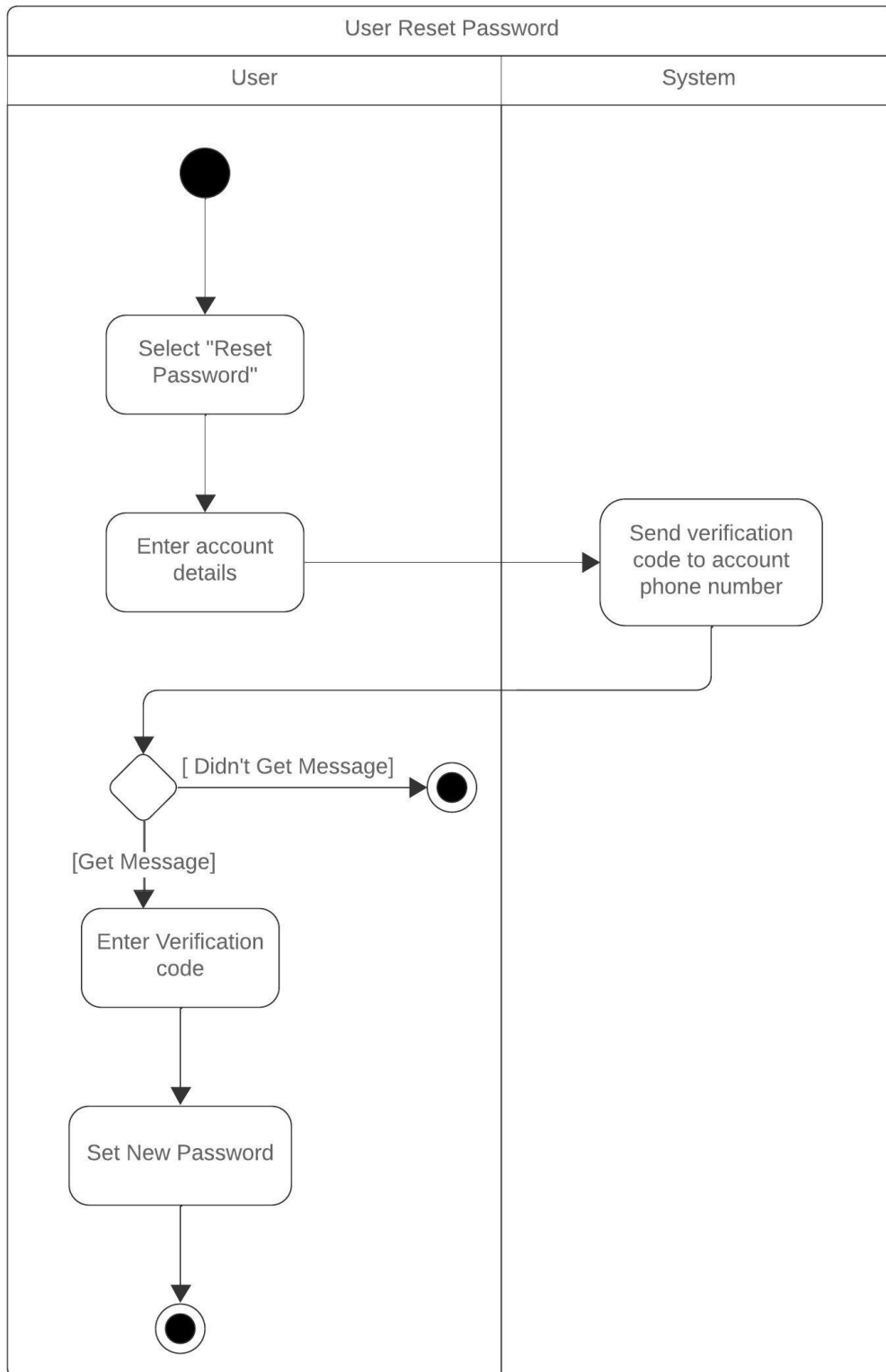
## 2.5 Check Balance



## 2.6 View Transaction History

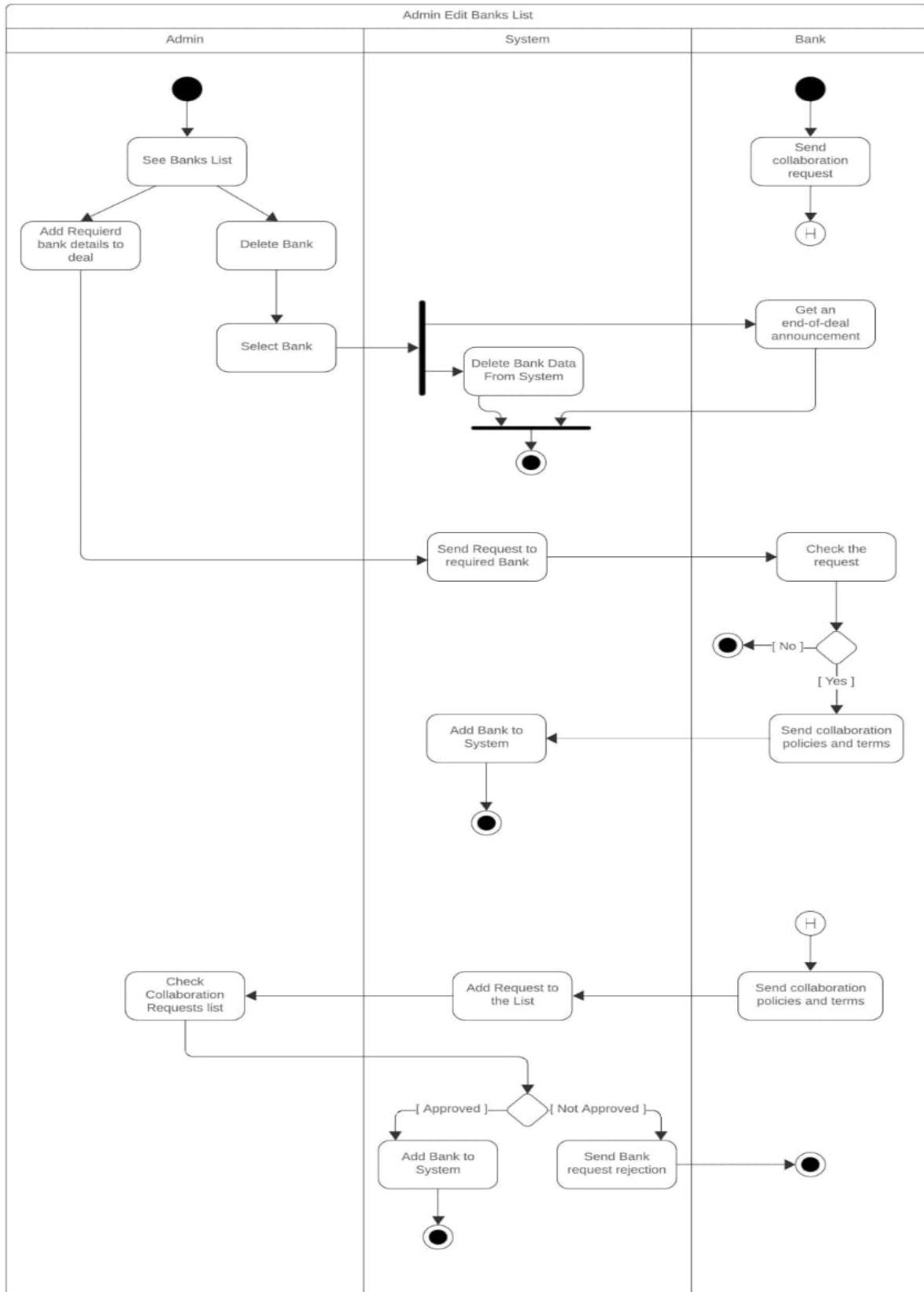


## 2.7 Reset Password

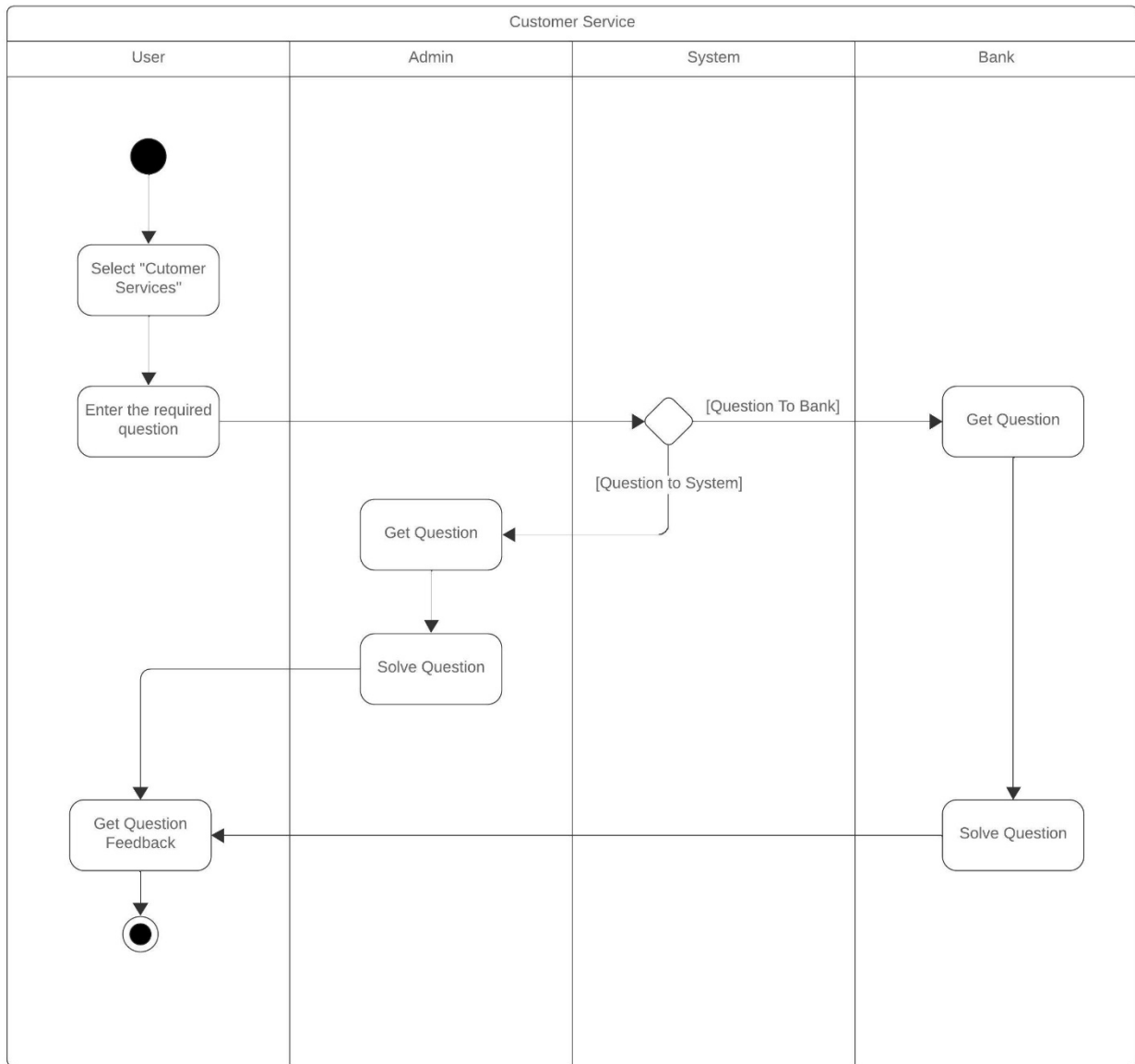




## 2.8 Admin Add Bank and modify

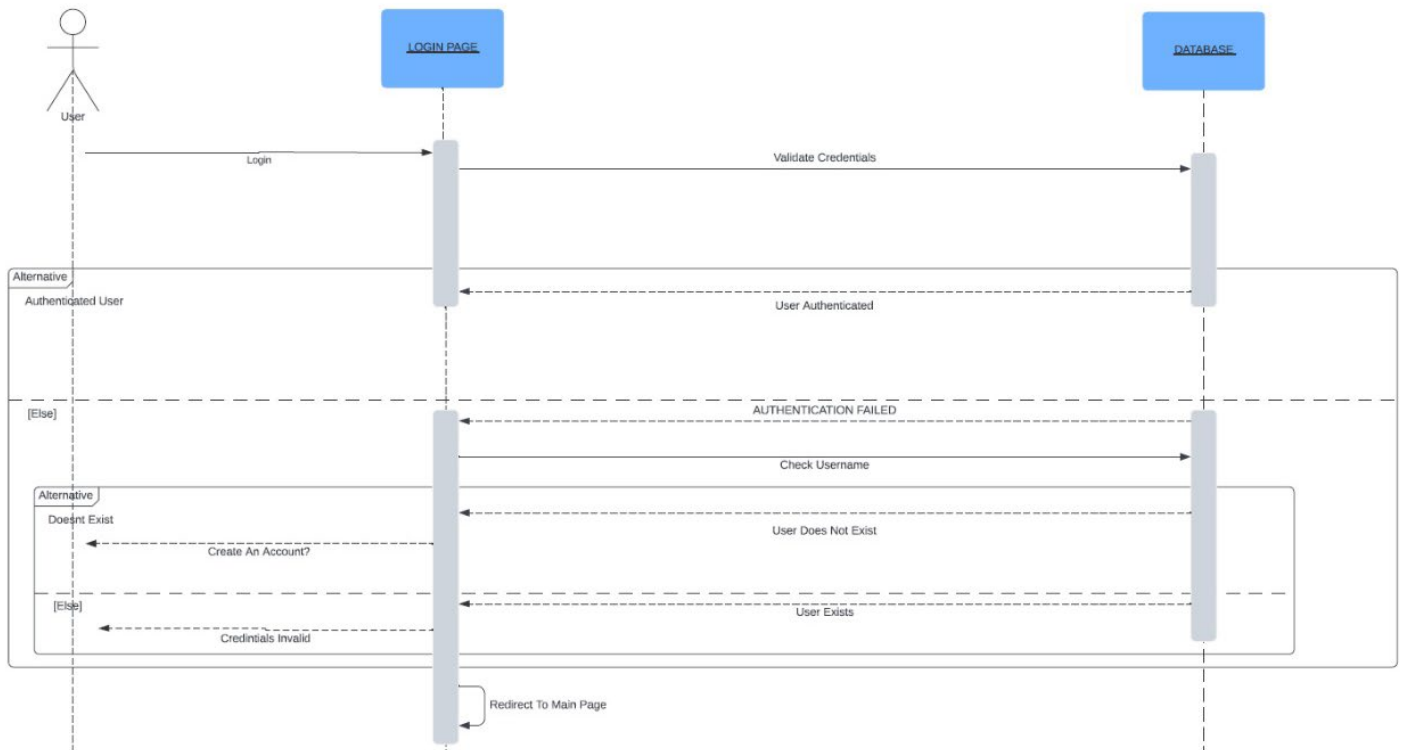


## 2.9 Customer Service and help

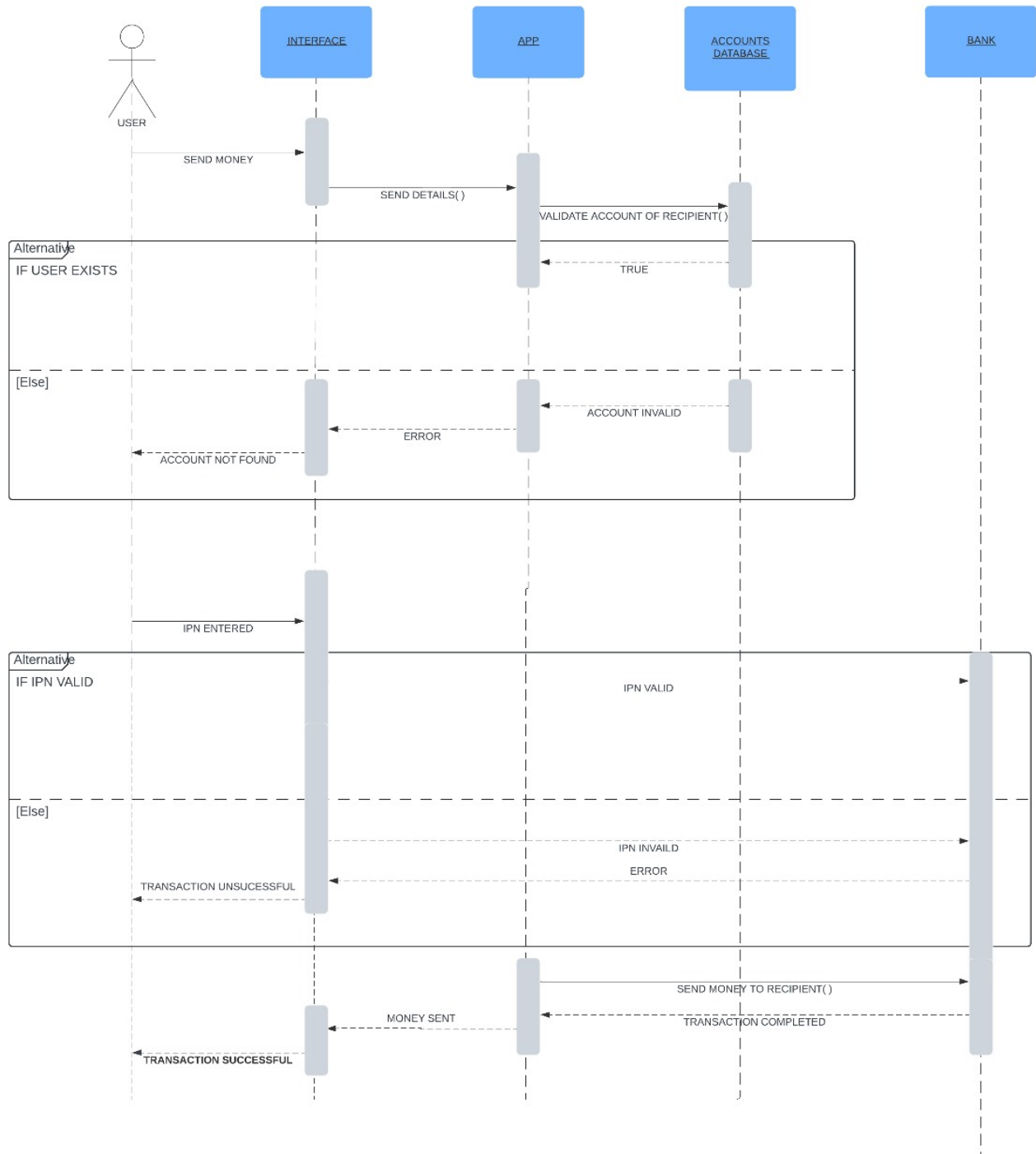


## 3. Sequence Diagram:

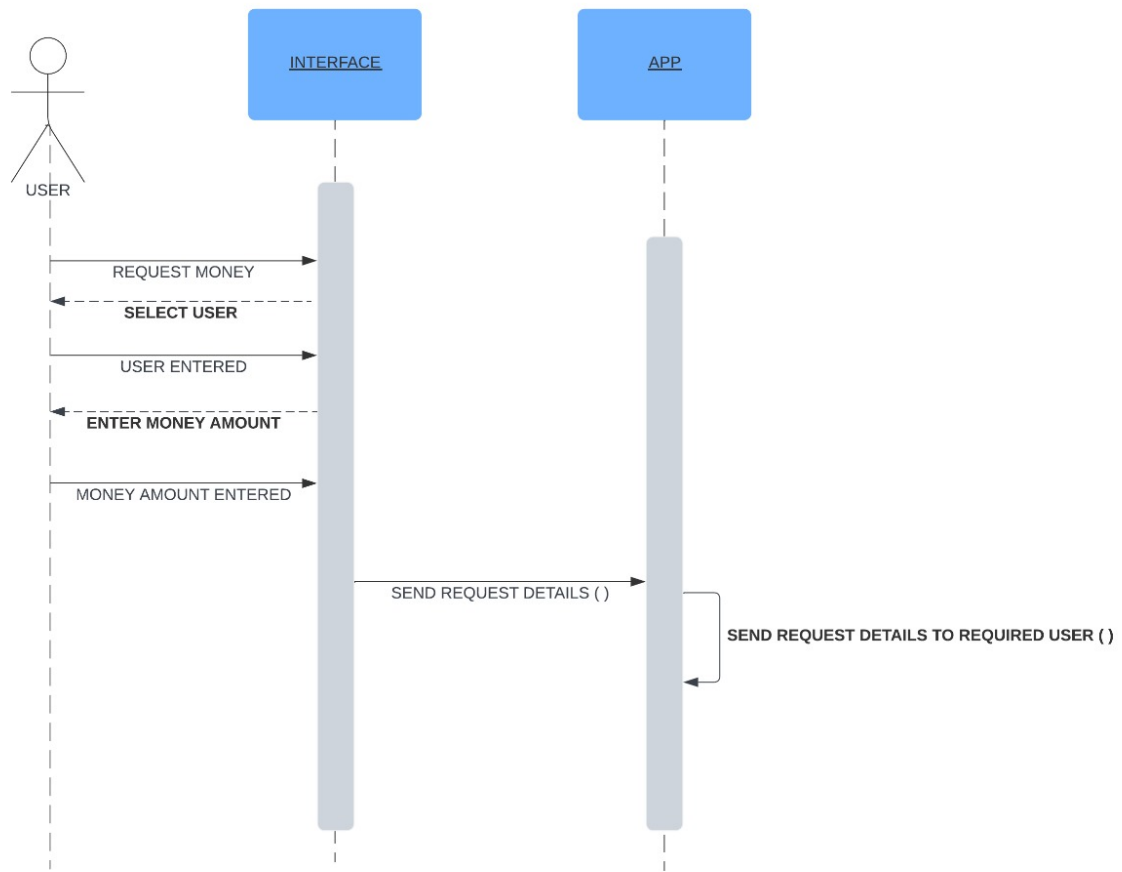
### 3.1 Login SSD



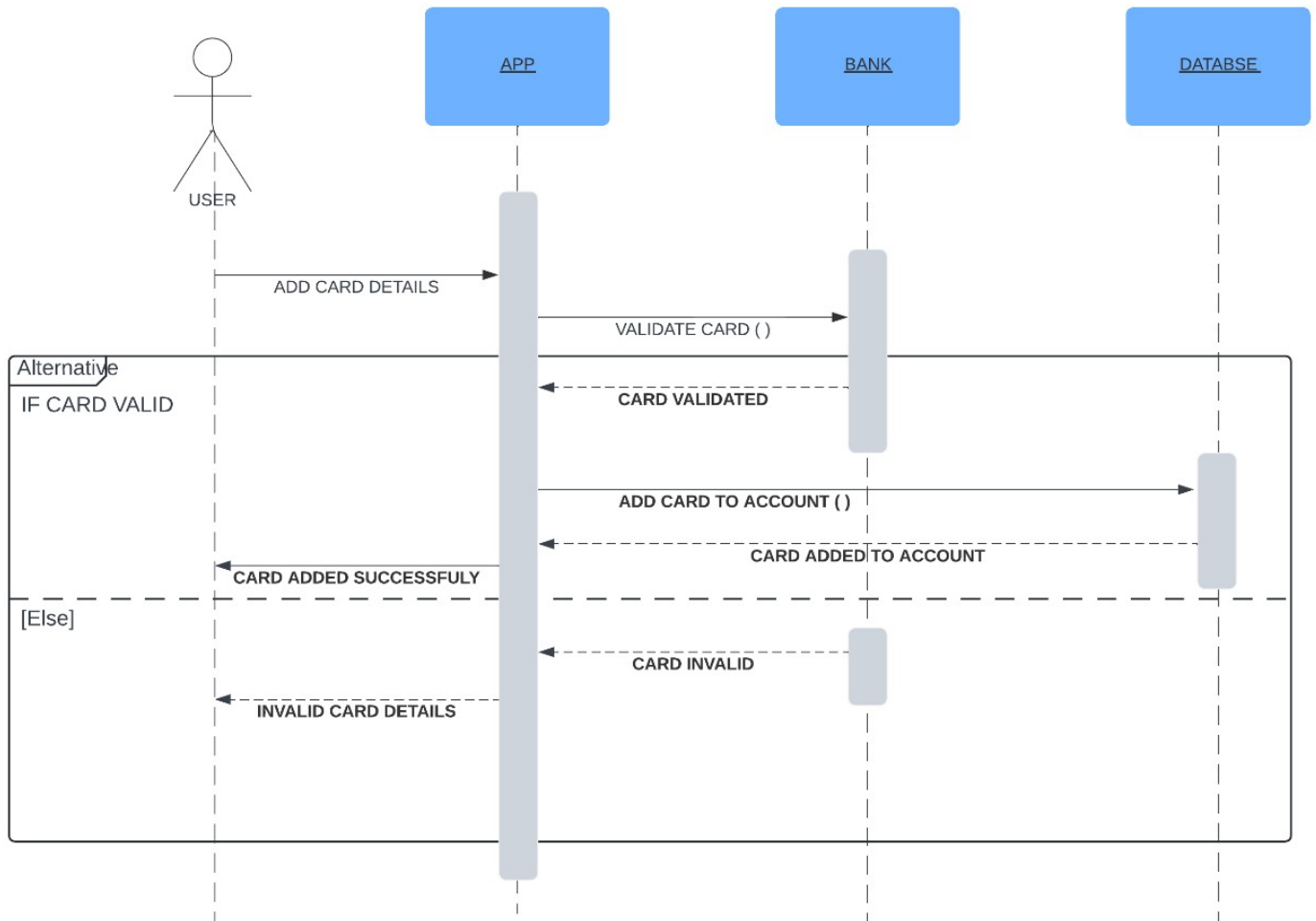
## 3.2 Send Money SSD



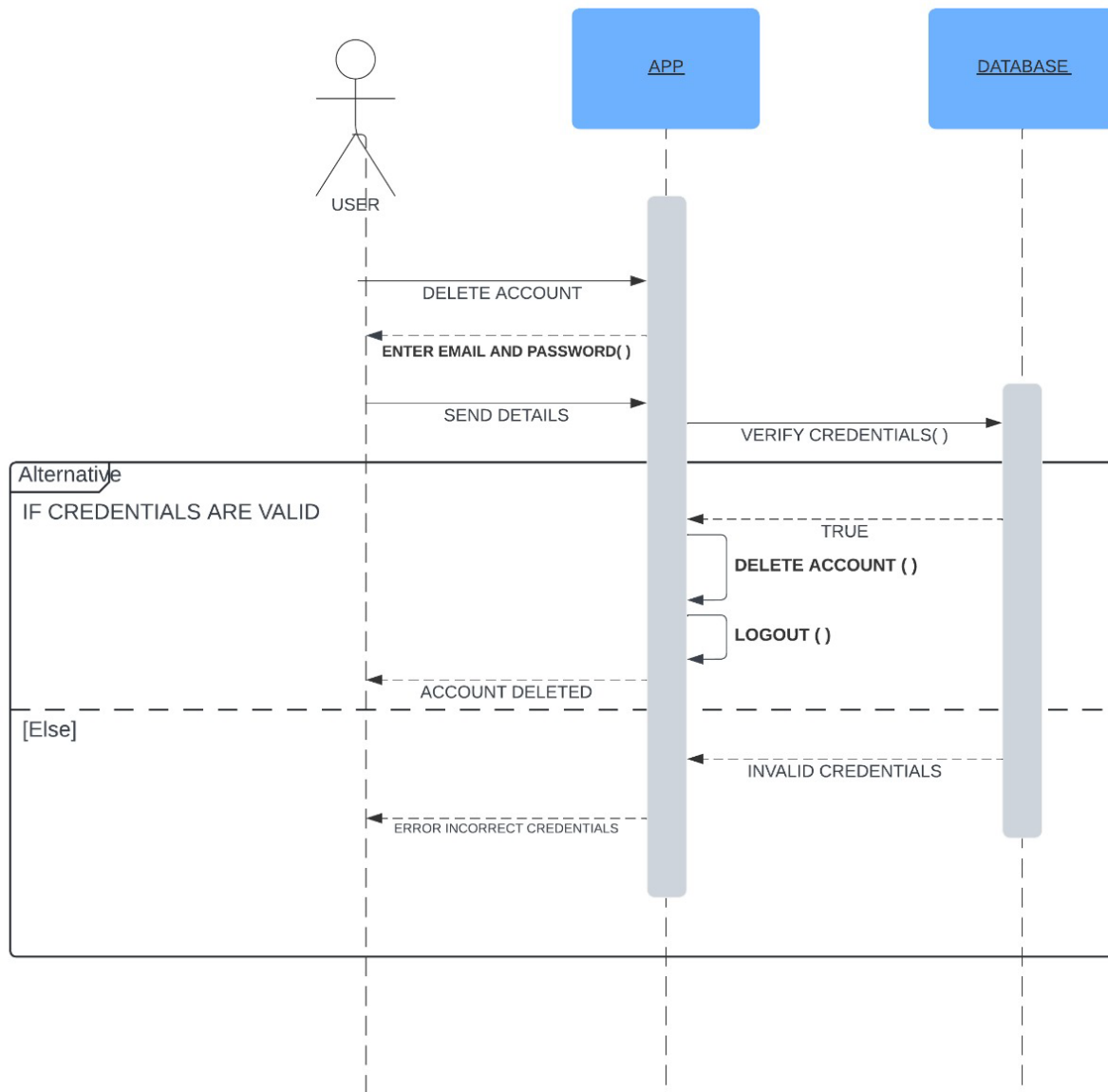
### 3.3 Request Money SSD



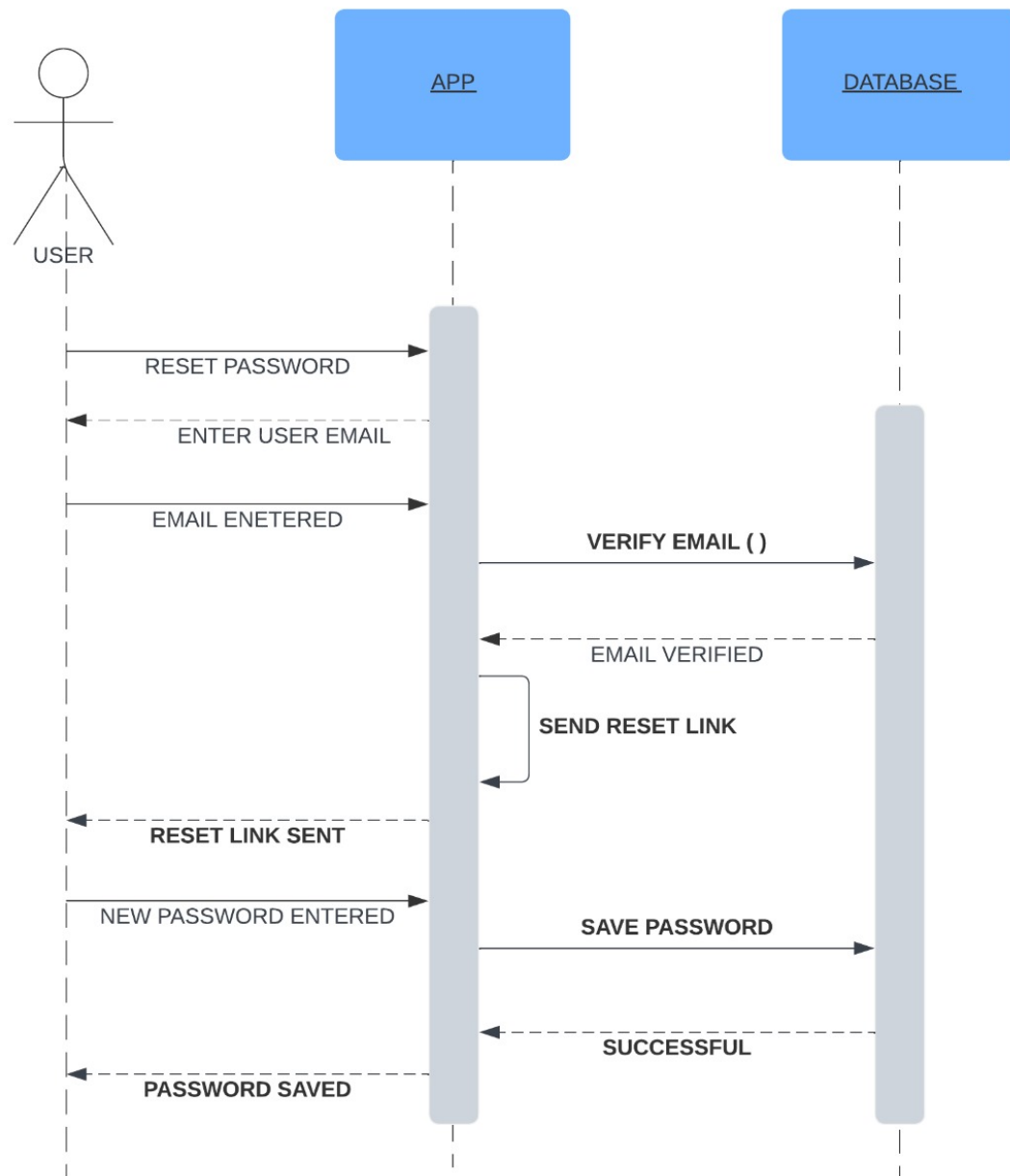
### 3.4 Add Card SSD



### 3.5 User Delete Account

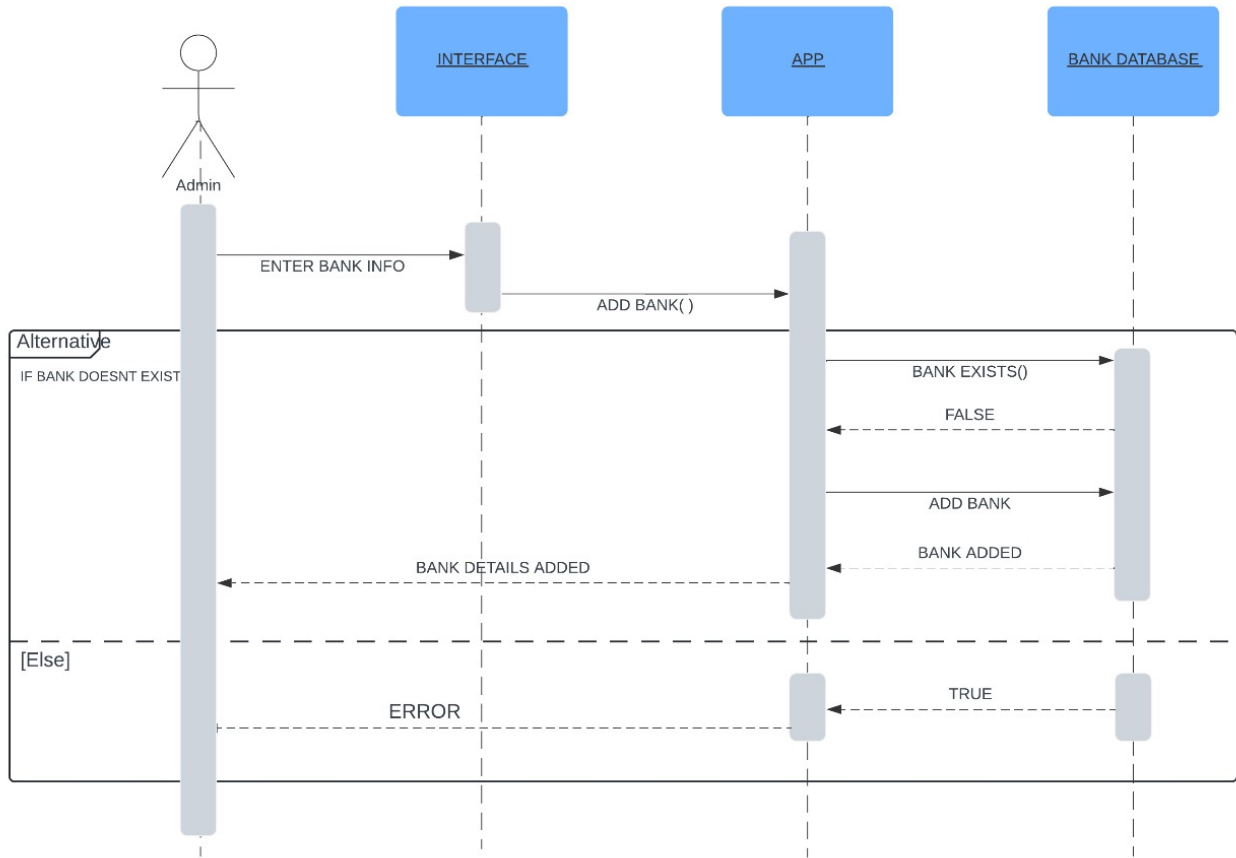


### 3.6 Reset Password SSD

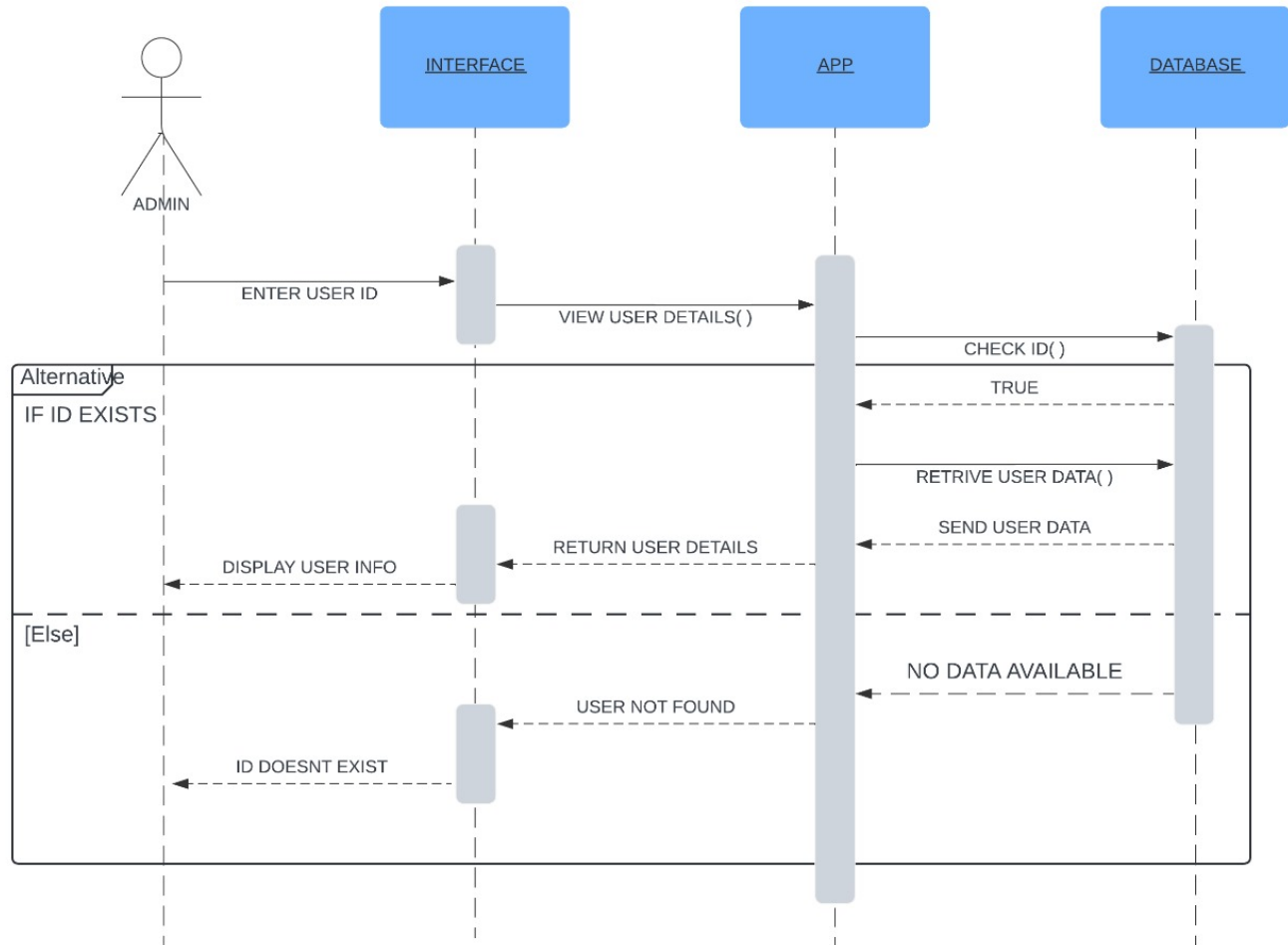




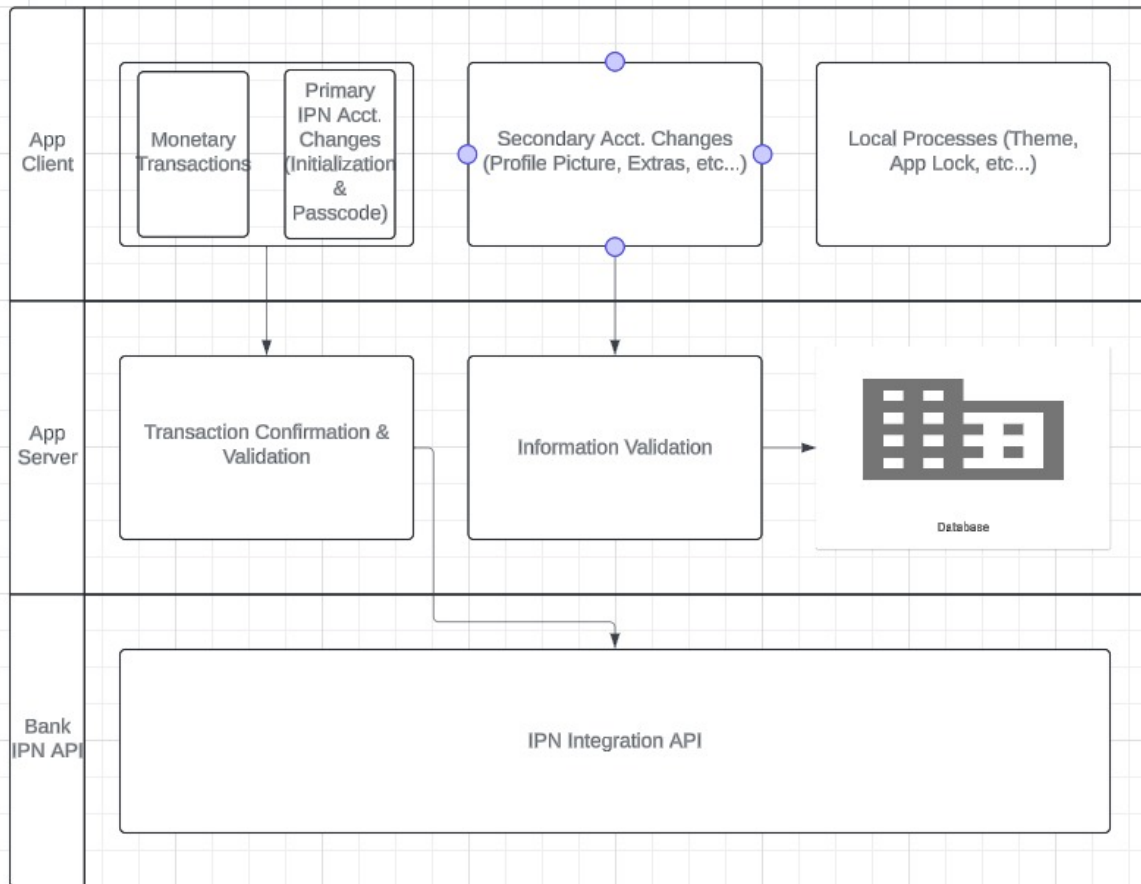
### 3.7 Admin Add Bank SSD



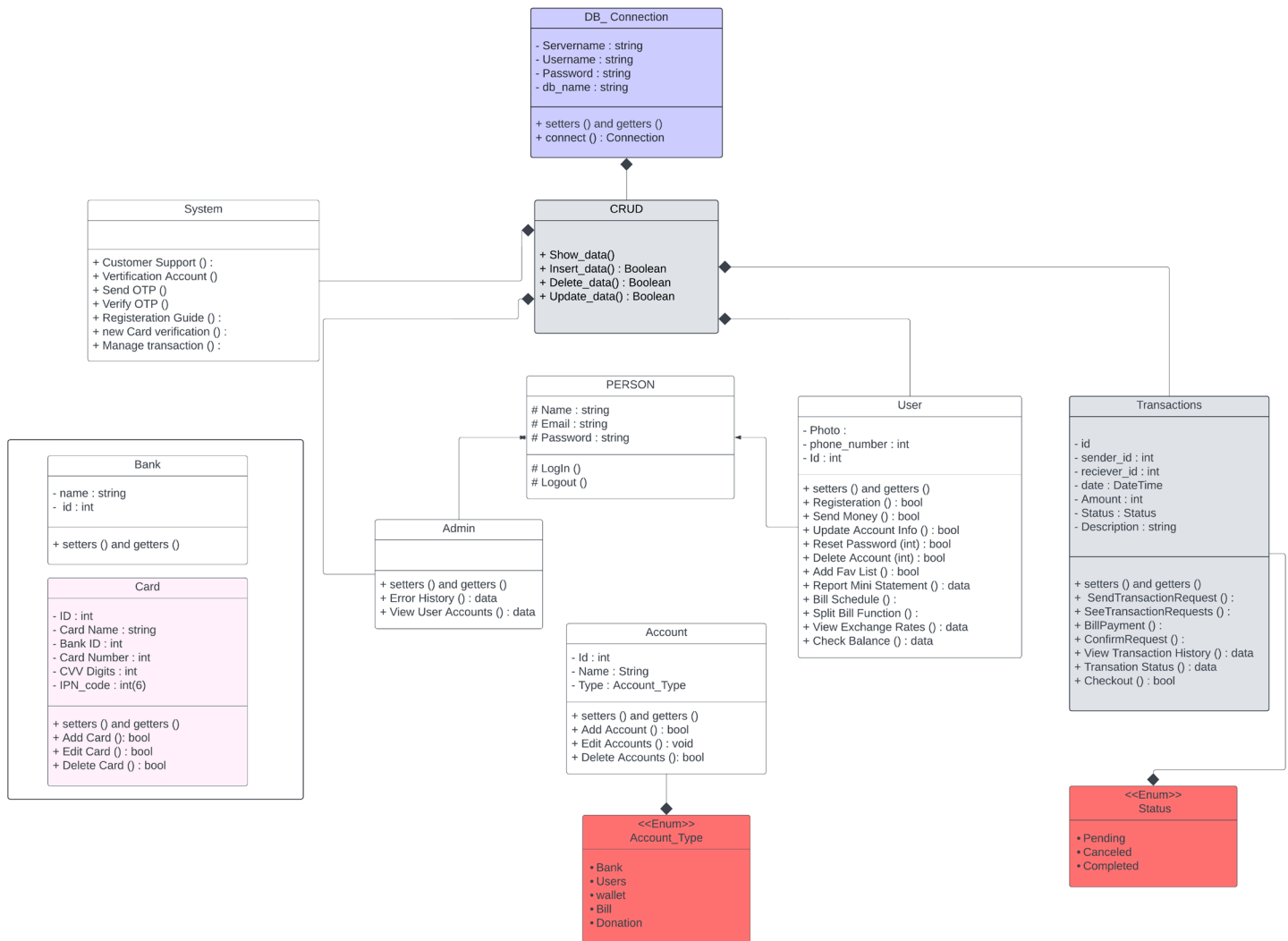
### 3.8 Admin View User Account



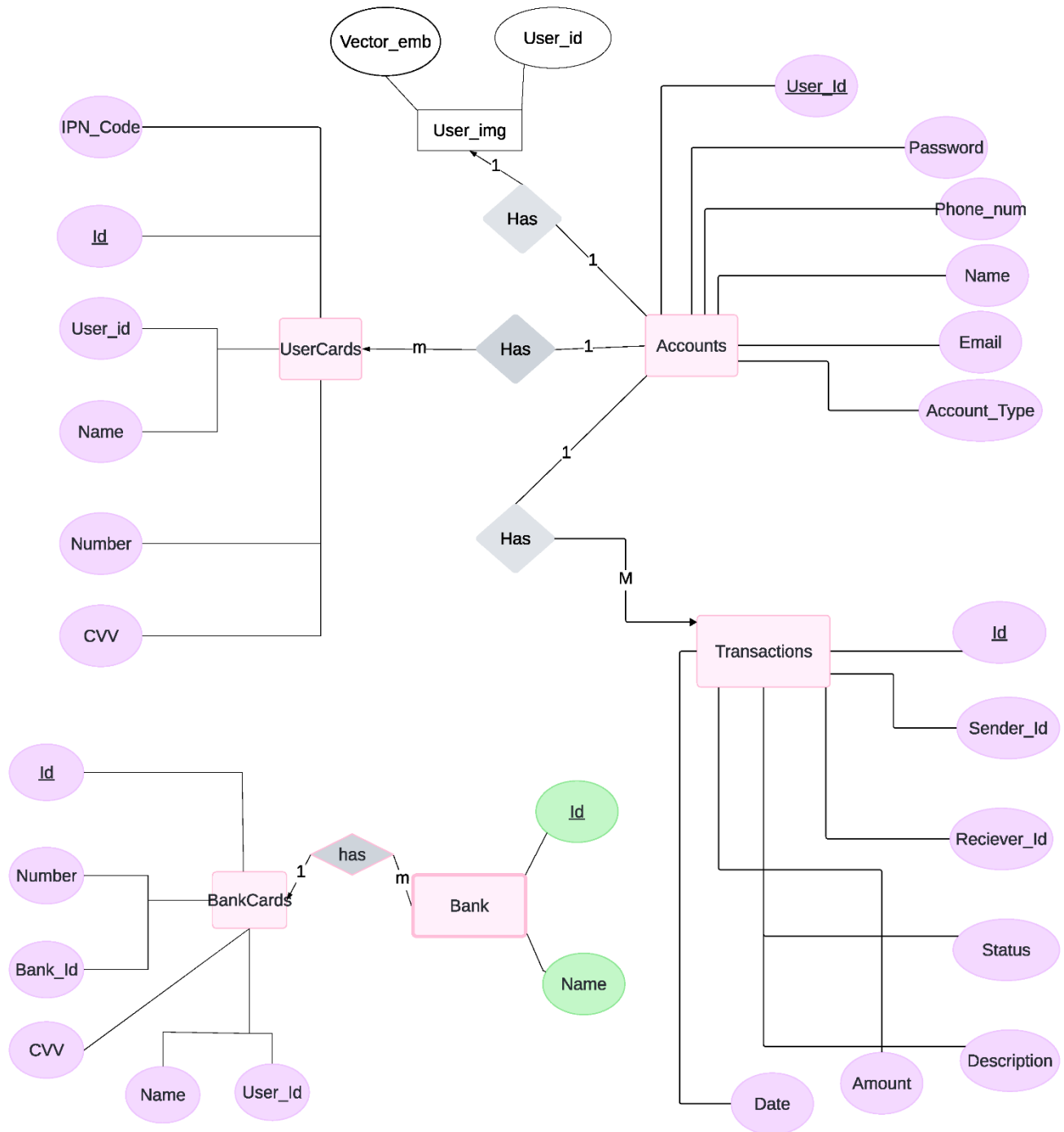
## 4. System Architecture:

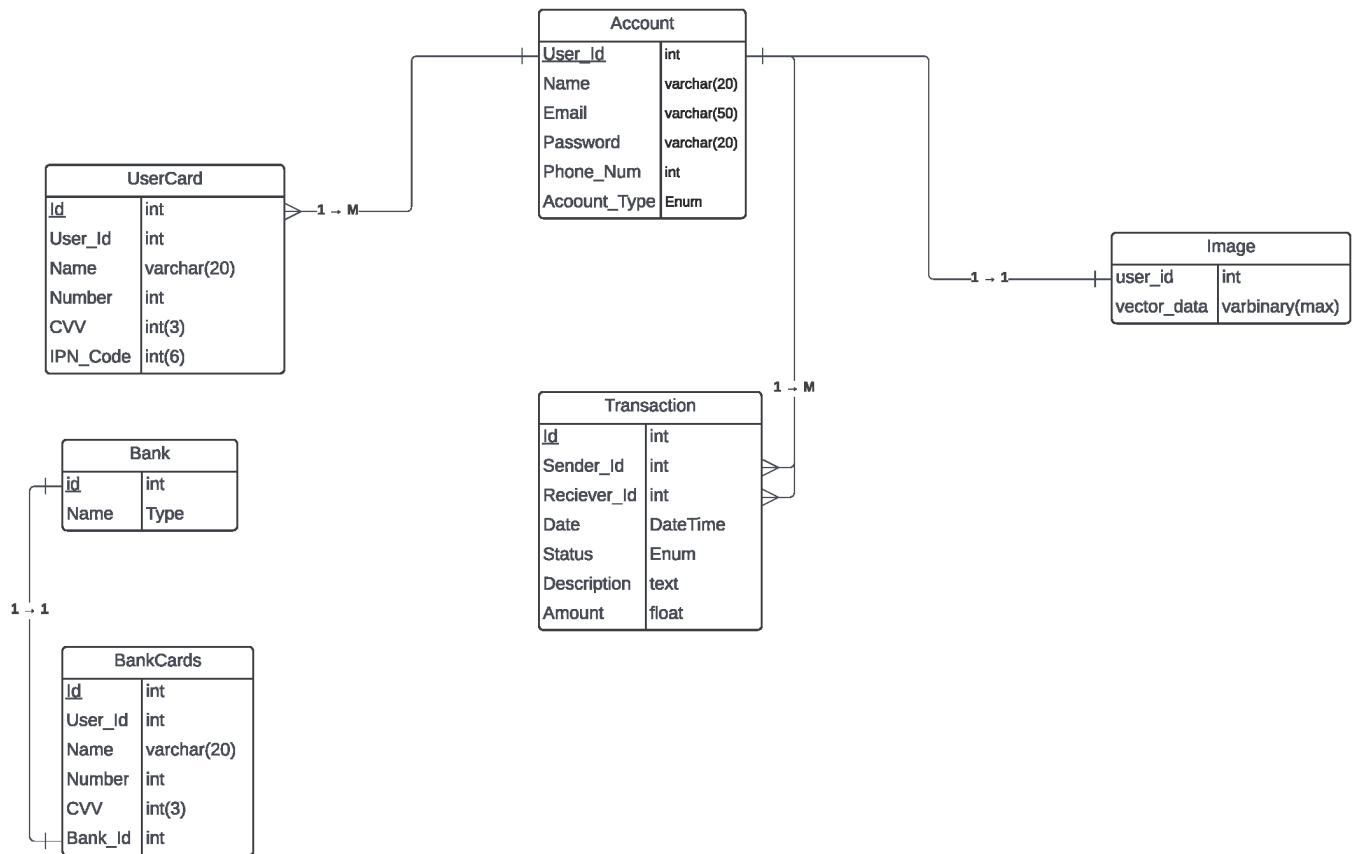


## 5. Class Diagram:

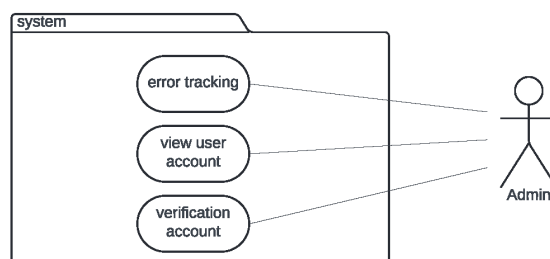
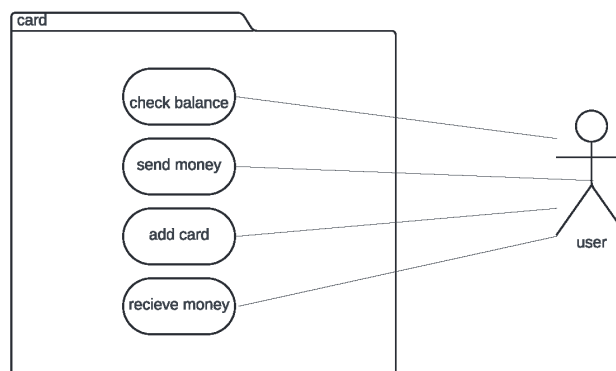
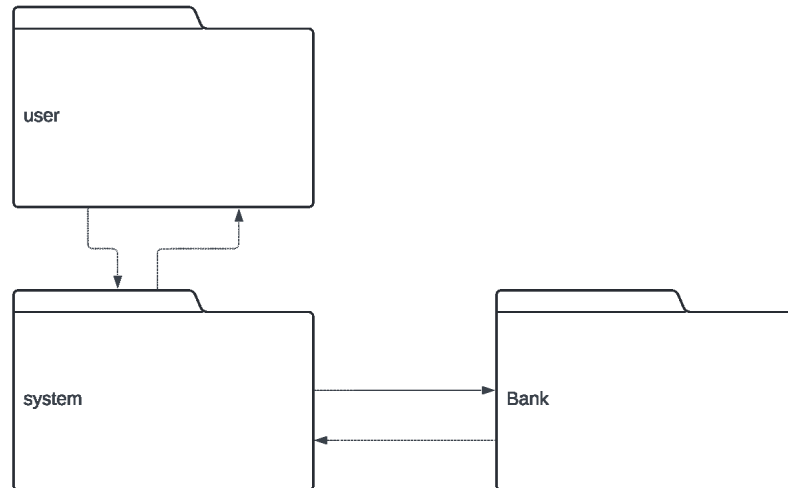


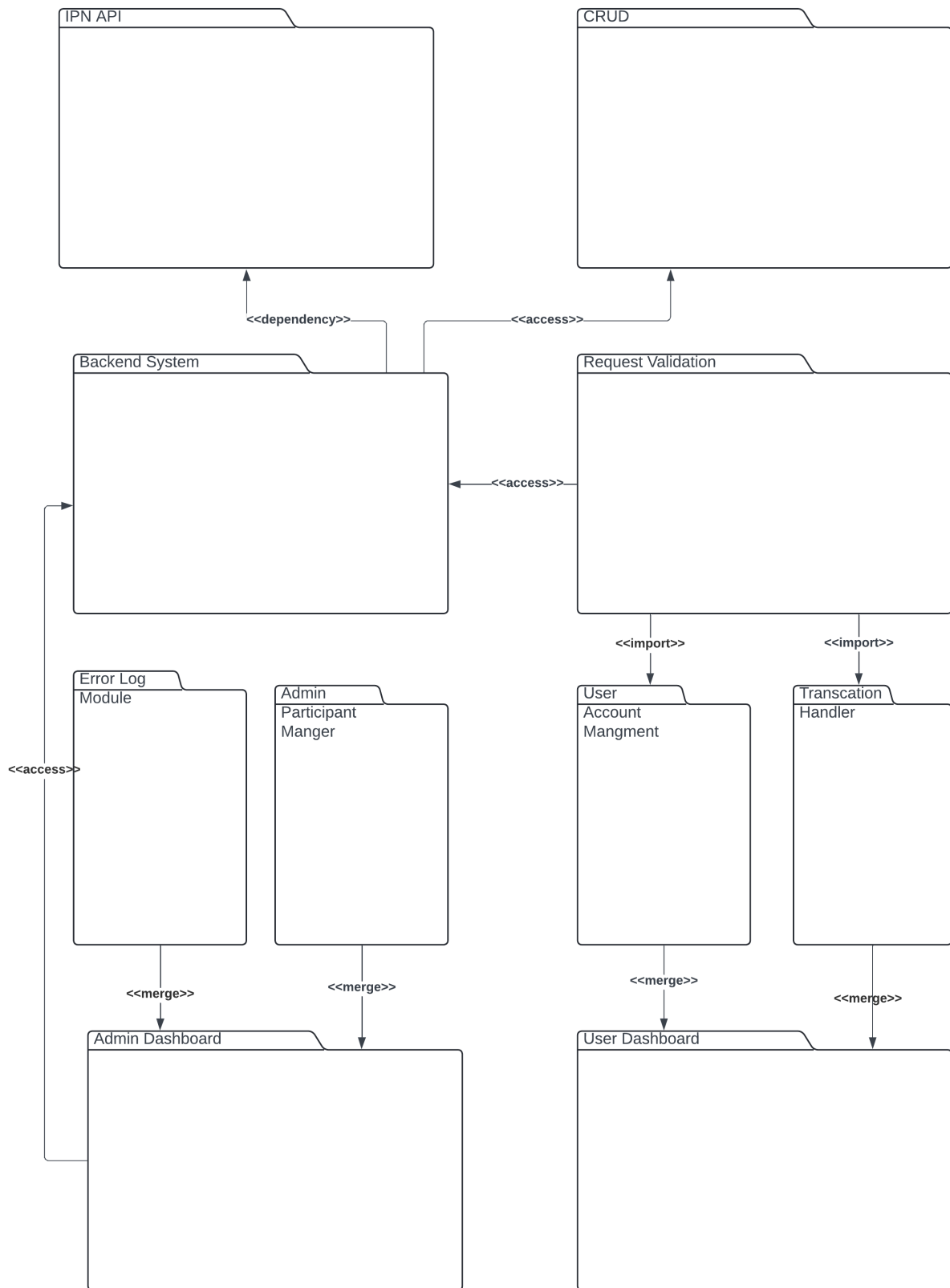
## 6. ERD Diagram:





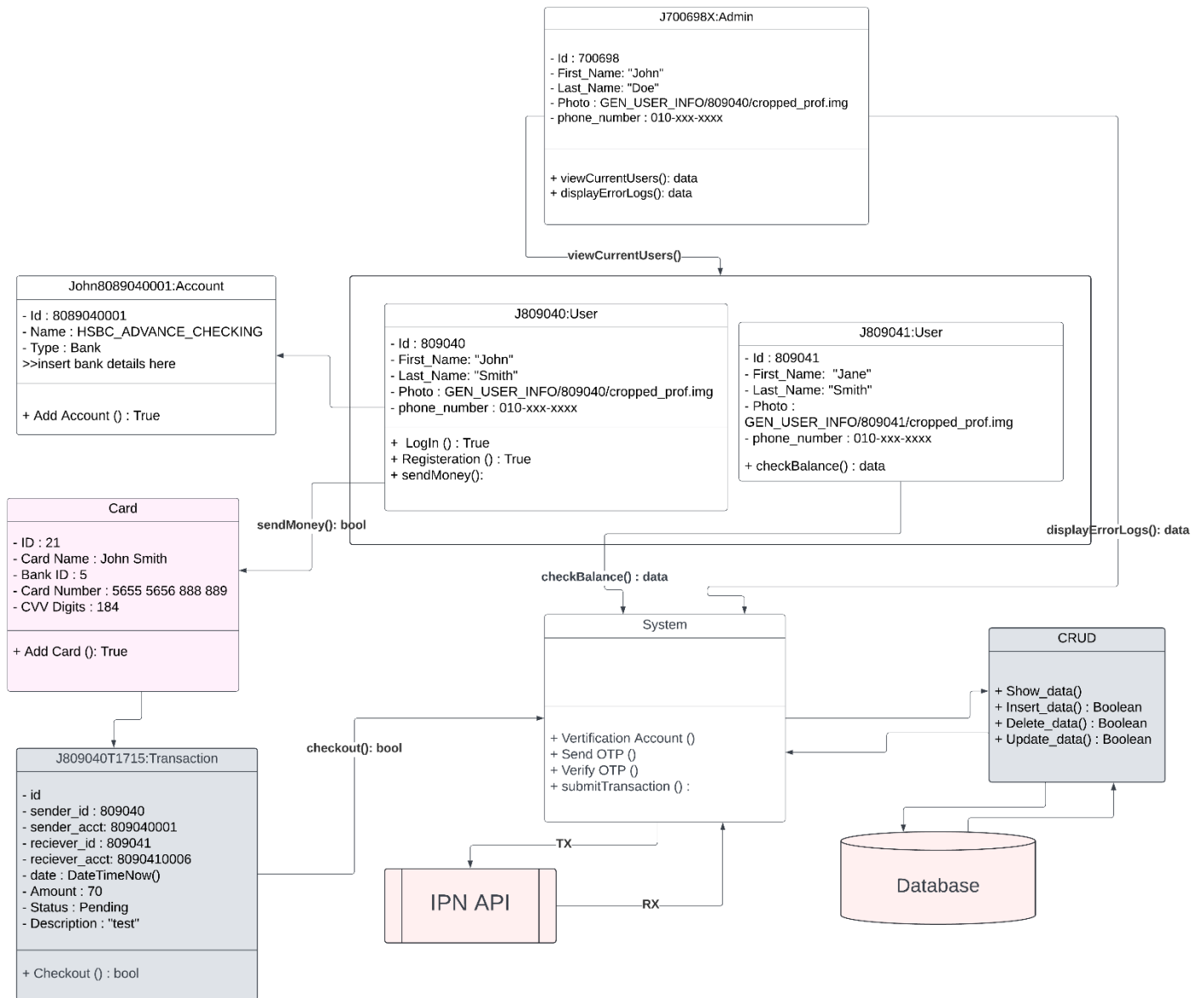
## 7. Package Diagram:







## 8. Object Diagram:



## 9. Snapshots: