Online Banking System

FINAL REPORT

Project Description

- An online banking system provides the following features to its client:

1. Create a new Bank Account providing Full Name, account password, initial amount of money to deposit then generate a unique Account ID for this user.

2. Login using the unique account ID and account password.

3. Check on Current Balance.

4. Deposit Cash to your account.

5. Withdraw Cash from your account.

6. Transfer Money to another account within the same bank.

7. Transfer Money to another account in another bank.

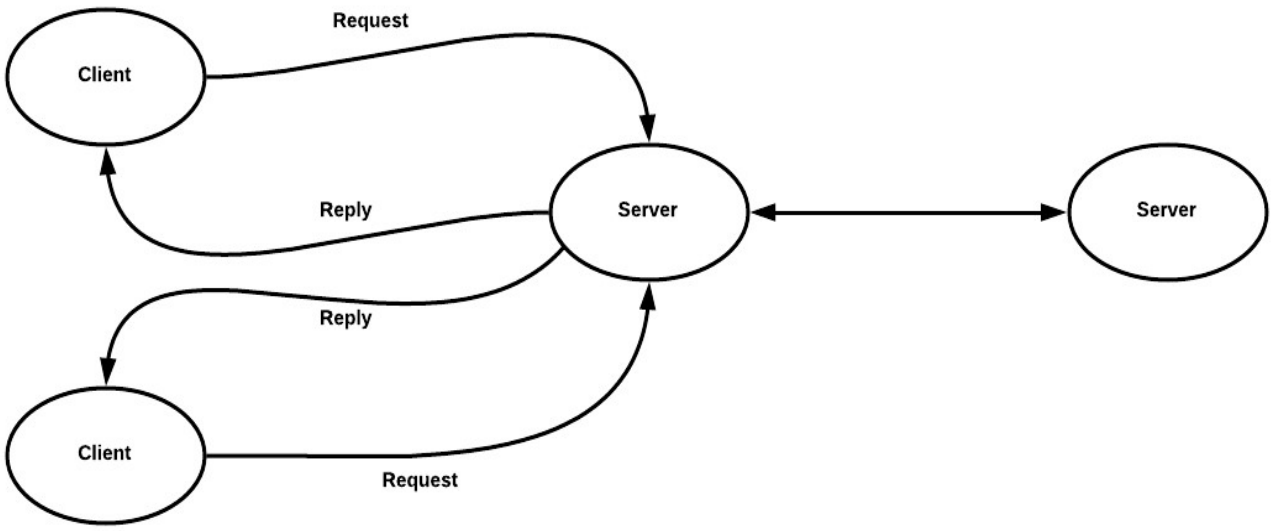
8. View Transaction History.

- Graphical user interface for the client side application.

- Database for the system.

- Encryption and Decryption techniques to save password in database.

SYSTEM DESIGN
System Architectural Model



" Client - Server "

" Peer - To - Peer "

Client – Server Model

"Server"
Communication Tasks

1. It creates socket with client.
2. It read client choice whether he wants to login or create account.
3. Login -> It reads ID and Password from the client.
4. Create Account -> It reads user information from the client.
5. It reads client choice that determine which service he wants to get.
6. Account Info -> It sends the balance to the client.
7. Deposit -> It sends the updated balance to the client.
8. Withdraw -> It sends the updated balance to the client.
9. Transfer money to another account -> It sends the updated balance to the client.
10. Transaction history -> It sends the history to the client.

Computational Tasks

1. It connects to database.
2. Login -> It checks in database if the Client enters the right ID and Password.
3. Create Account It saves user information in database.
4. Account Info -> It get the balance from database.
5. Deposit -> It get the balance.
6. Withdraw -> It checks the amount of money. if it's valid, it updates the balance.
7. Transfer money to another account -> It checks the account ID and amount of money. if they are valid, it updates the balance of the two users.
8. It saves the history in database.
9. Transaction history -> It gets the history from the database.

“Client”

Communication Tasks

1. It invokes the server.
2. It sends his choice to choose login/create account.
3. Login -> It sends account ID and his password.
4. Create Account -> It sends user information then read account ID.
5. It sends choice that determine which service he wants to get.
6. Account Info -> -> It reads the balance from server.
7. Deposit and Withdraw -> It sends the amount of money then reads the updated balance from server.
8. Transfer money to another account -> It sends the account ID and the amount of money then reads the updated balance from the server.
9. Transaction history -> It reads the history from the server.
10. It ends connection.

Computational Tasks

1. It provides gui to the user.
2. Login -> It takes the ID and Password from user.
3. Create Account -> It takes the user information from user then checks if the two passwords are matched and checks if the amount of money is valid (>=0).
4. Deposit and Withdraw -> It takes the amount from user and checks if it’s valid (>=0).
5. Transfer money to another account -> It takes the account ID and amount of money from user then checks if it’s valid (>=0).
6. It informs the user if there is any error.

Peer – To – Peer Model

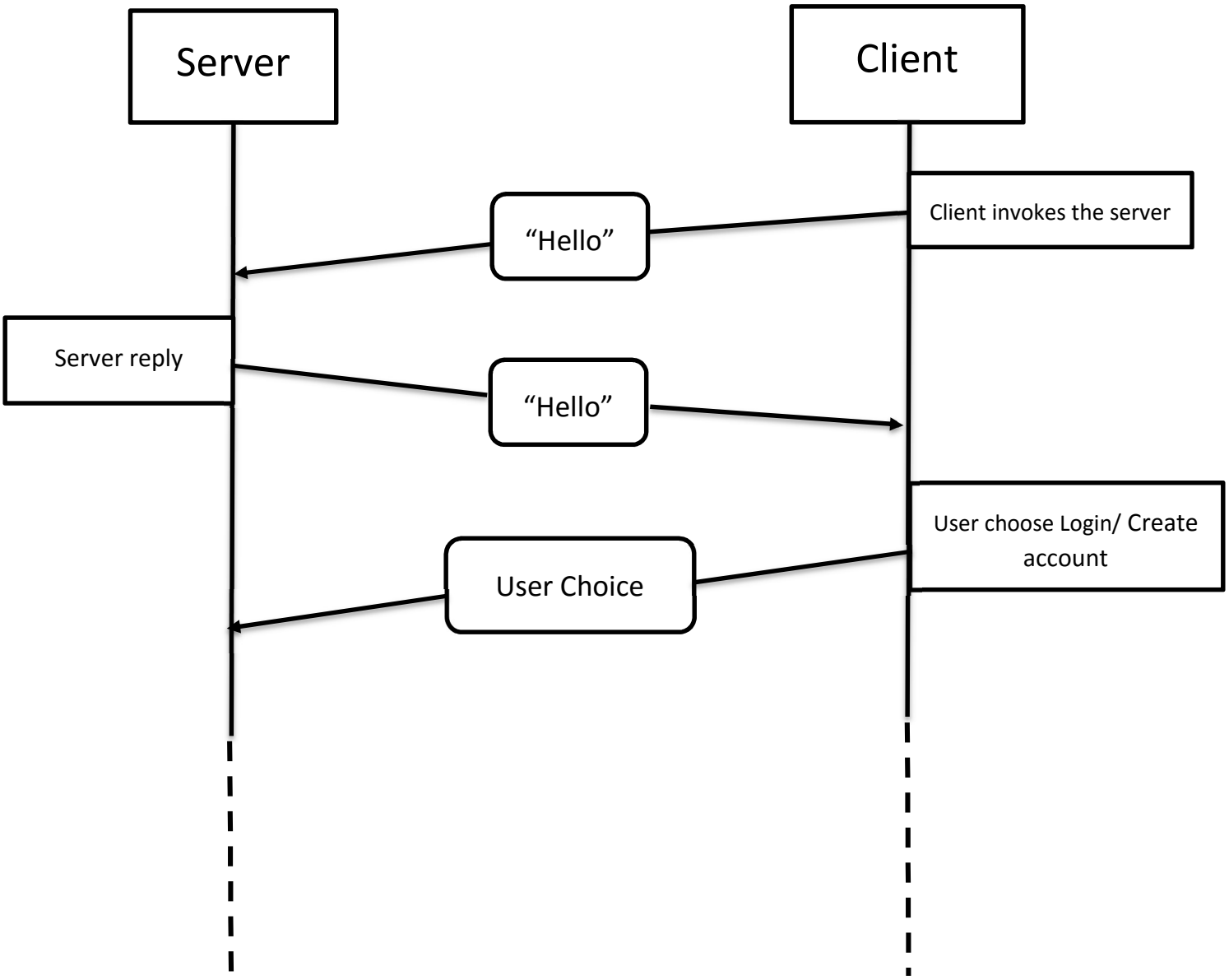
Communication Tasks

1. It invokes another Server.
2. It sends or reads the account ID of the client and the amount of money to be transferred.
3. It sends confirmation message to another Server (That transfer is done).
4. It ends the connection.

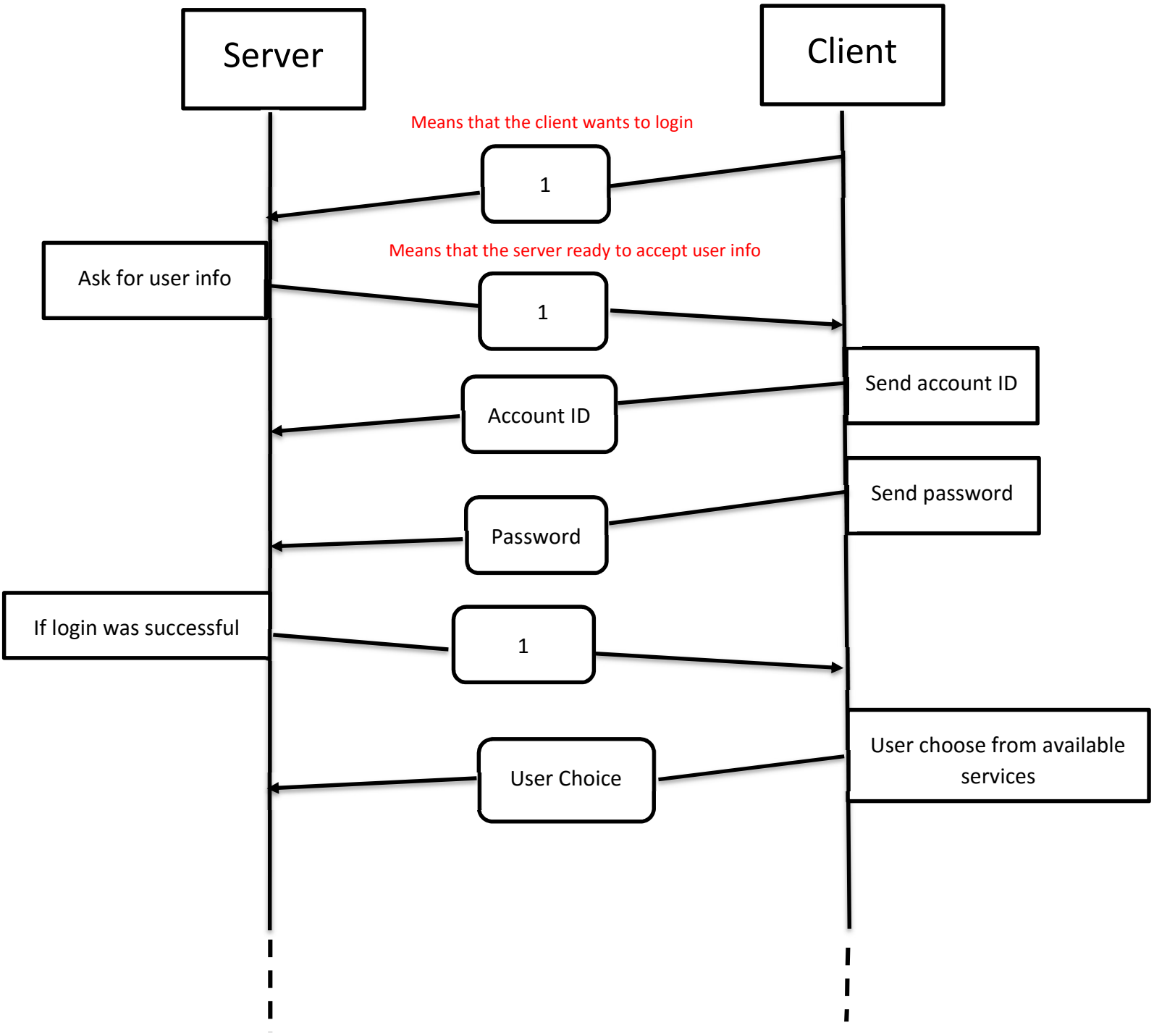
Computational Tasks

1. It checks for the account ID in database.
2. If account ID is valid it updates the balance in database.

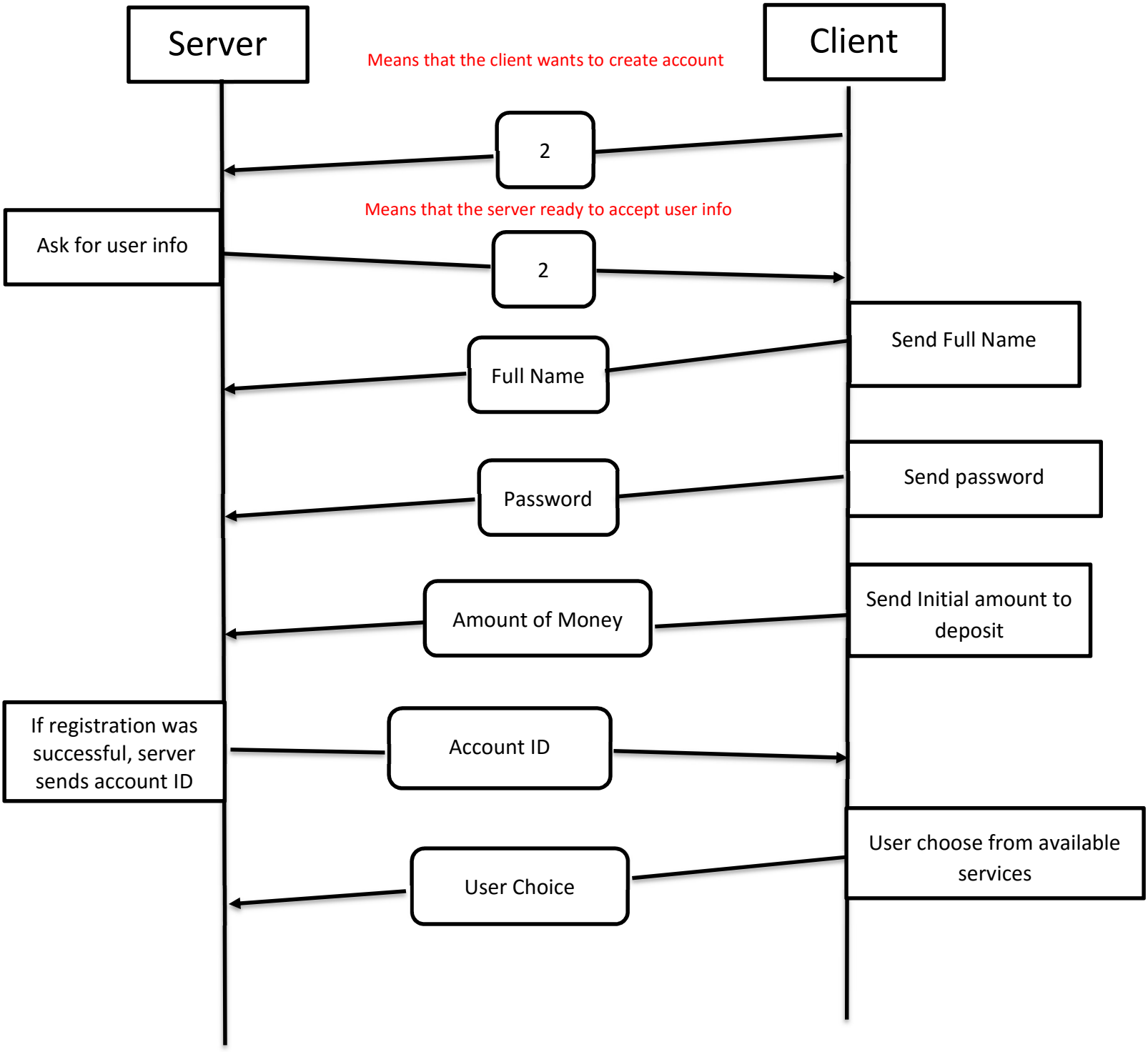
The Application Level Protocol



1. Login Case:

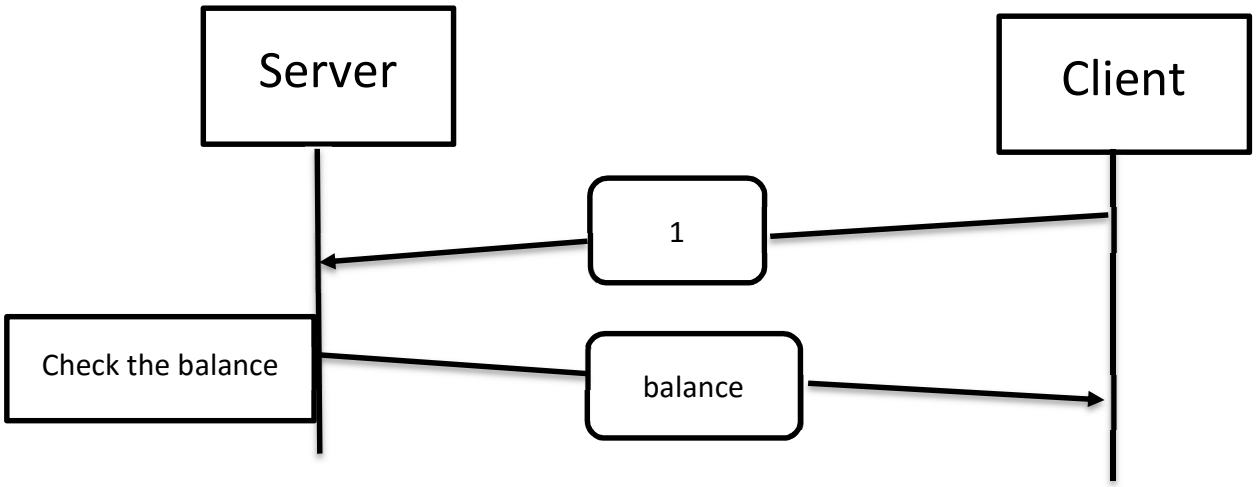

Server
Client
Means that the client wants to login
1
Means that the server ready to accept user info
Ask for user info
1
Send account ID
Account ID
Send password
Password
If login was successful
1
User choose from available services
User Choice

2.Create Account Case:

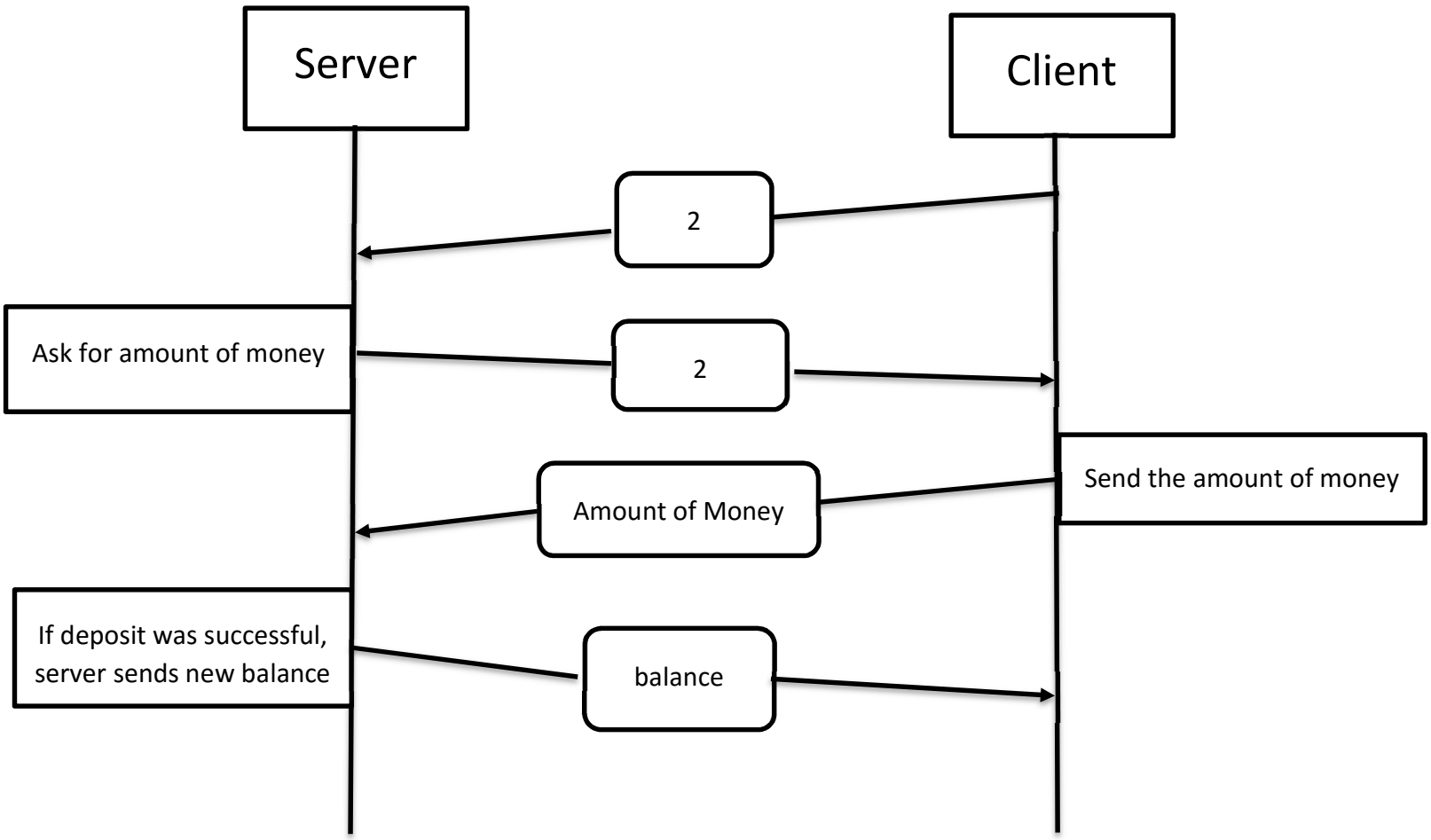


Services

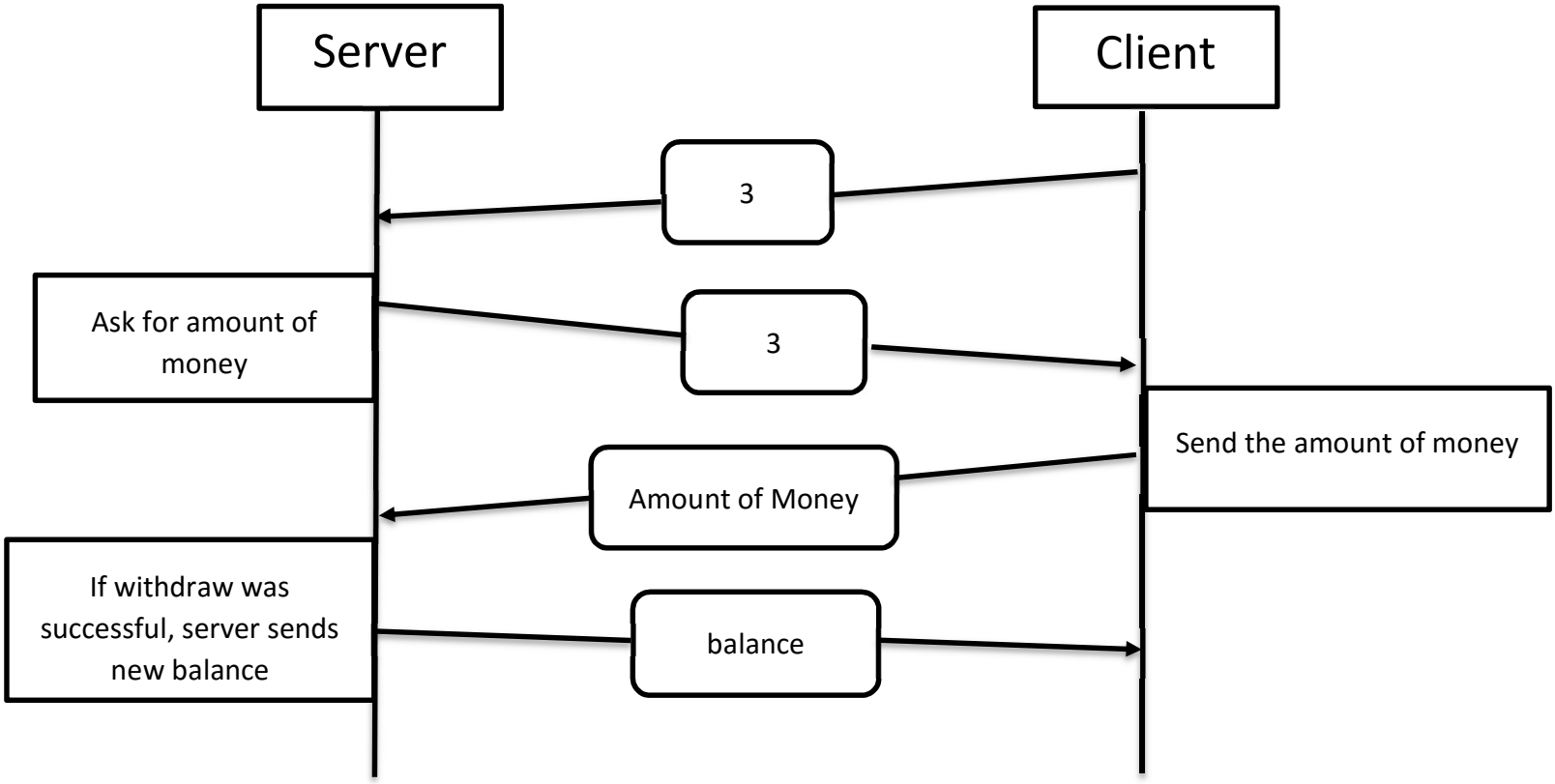
1) Check on current balance:



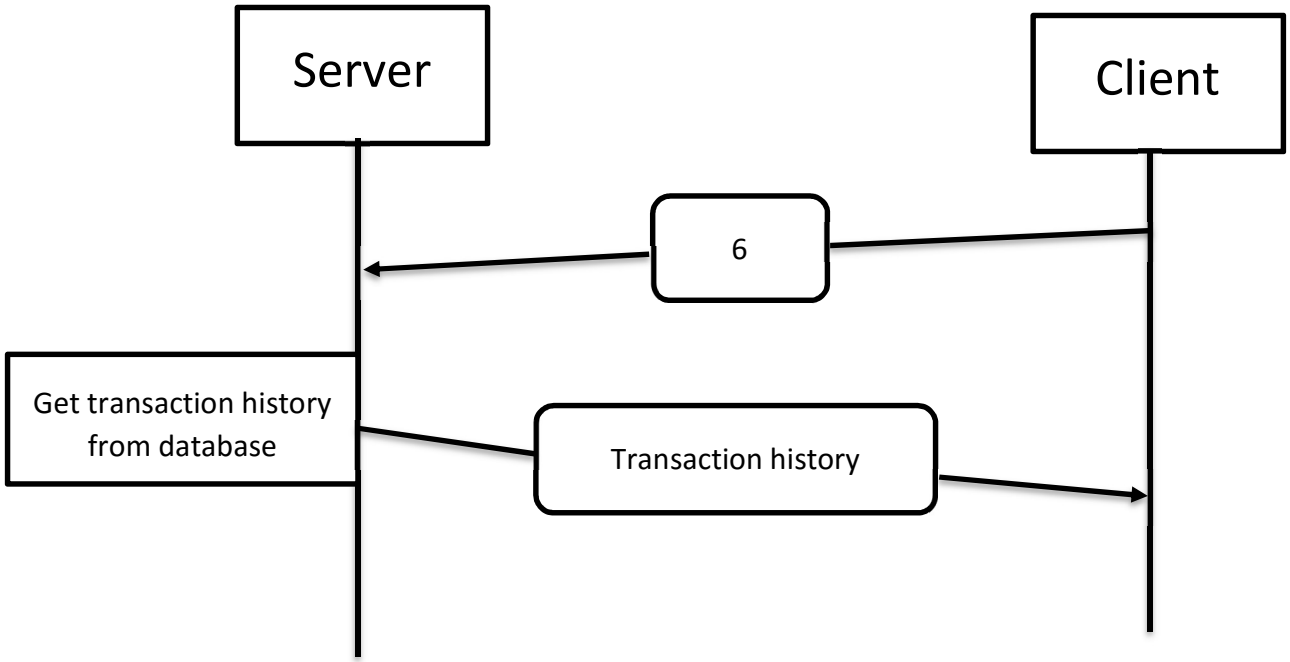
2) Deposit:



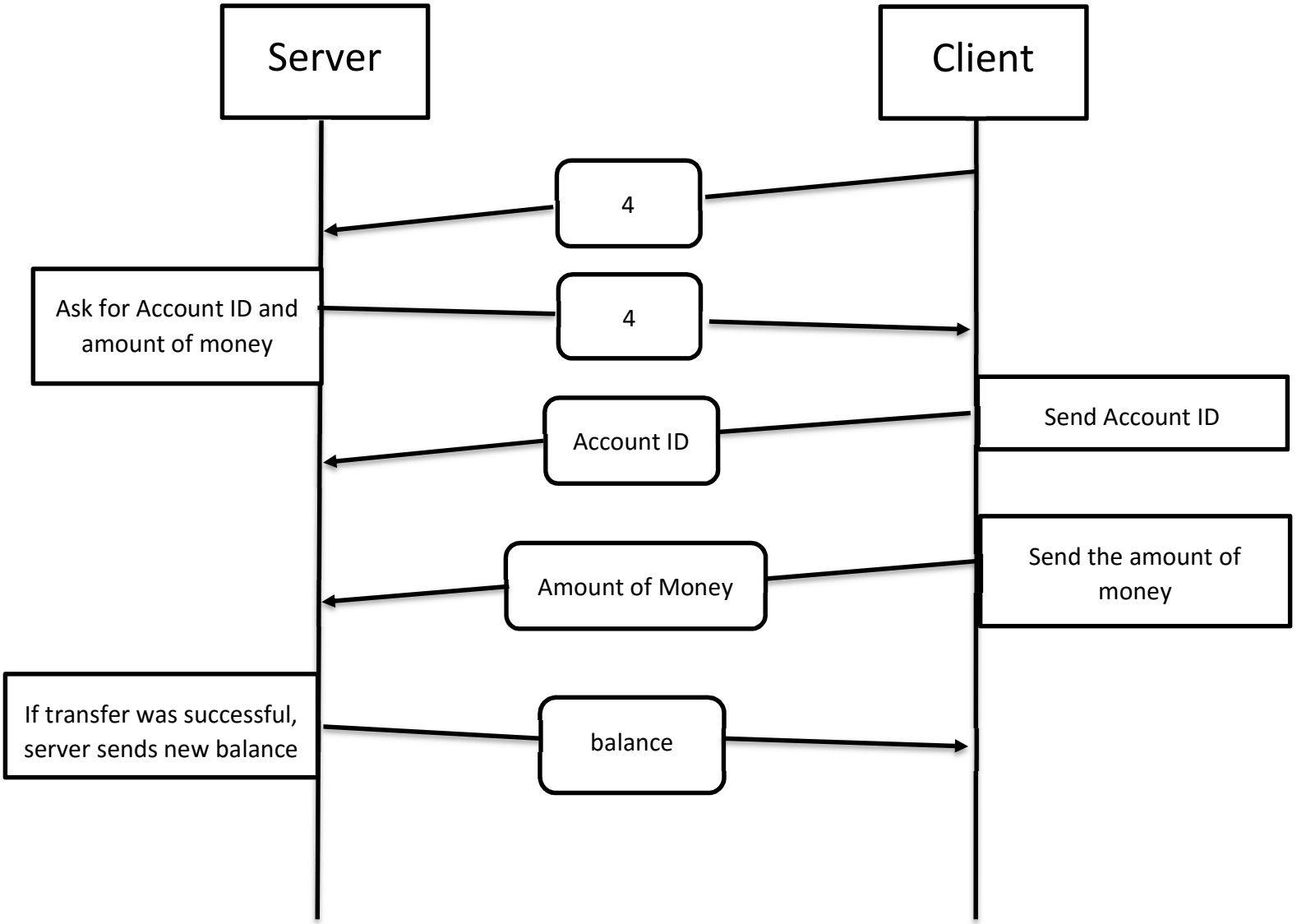
3) Withdraw:



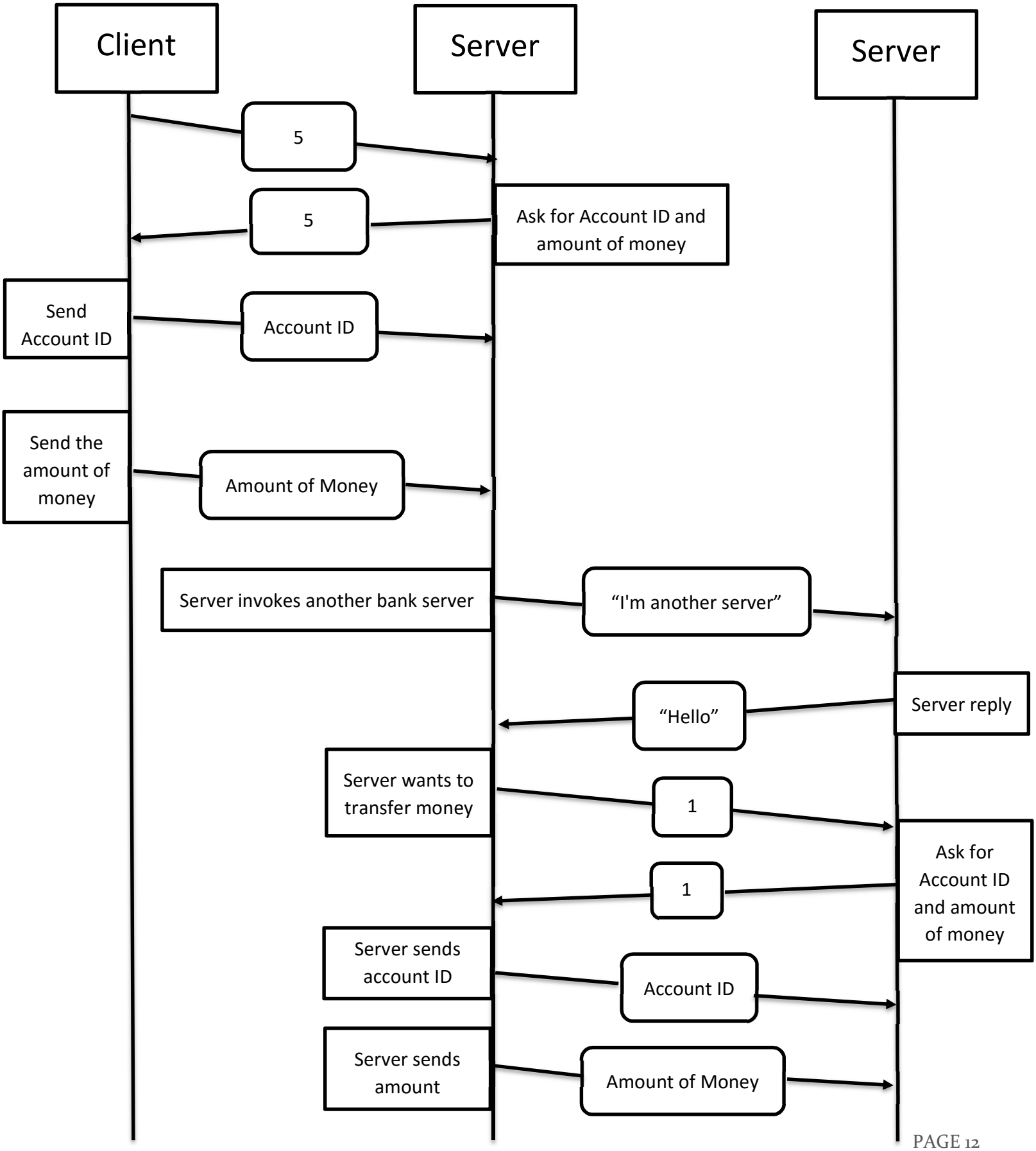
4) View transaction history:

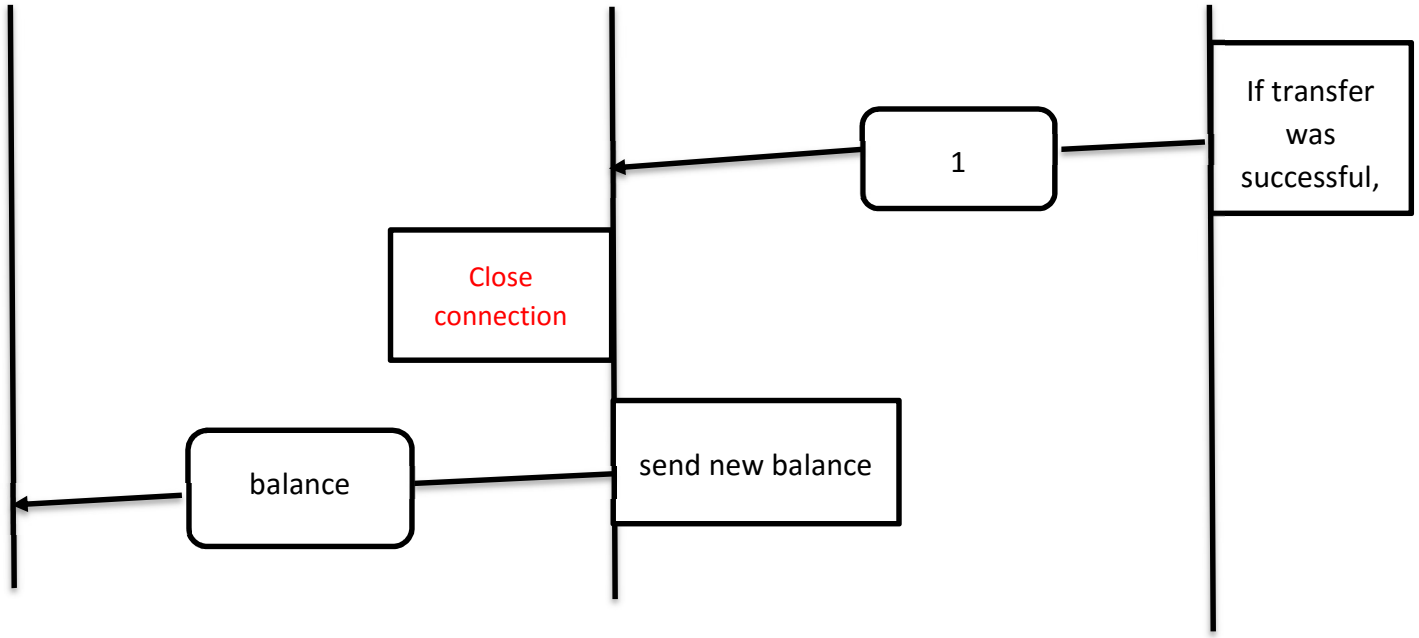


5) Transfer Money to another account within the same bank:

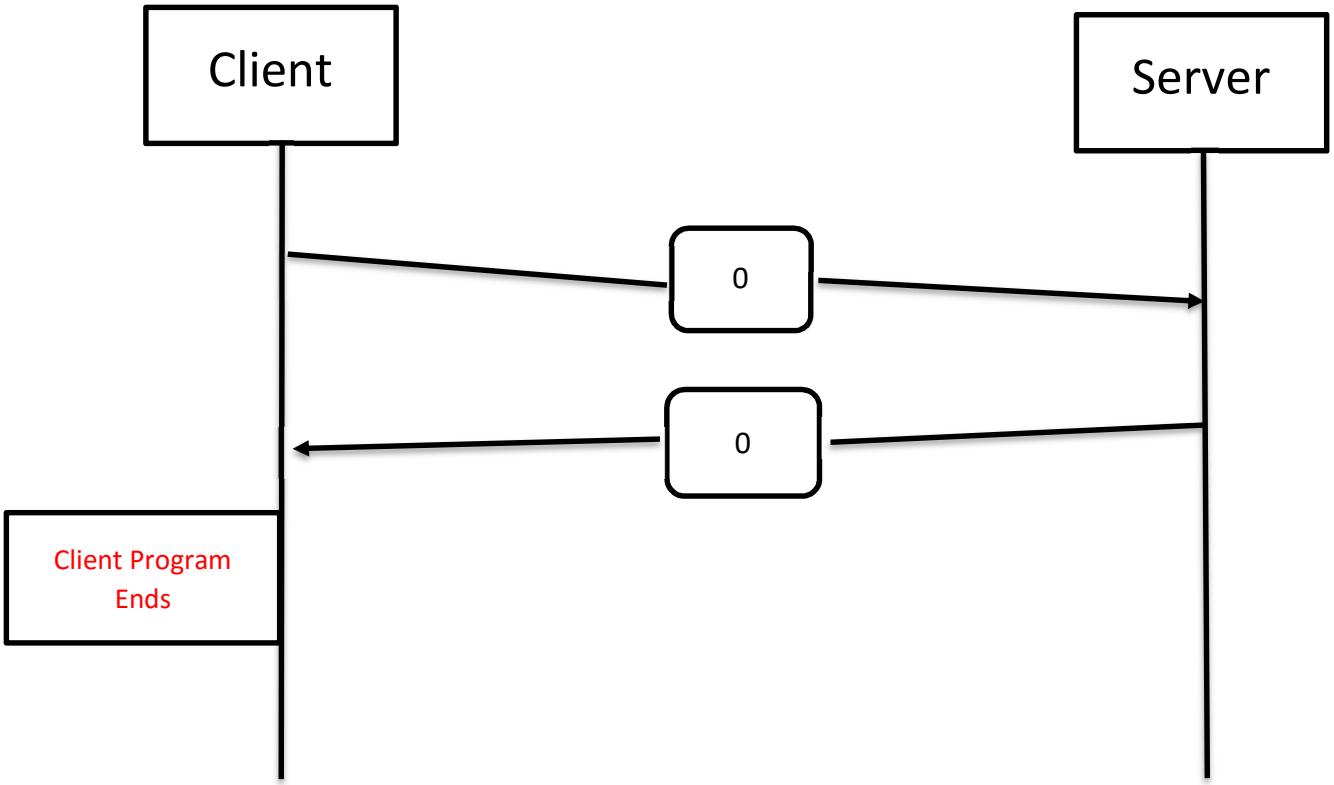


6) Transfer Money to another account in another bank:



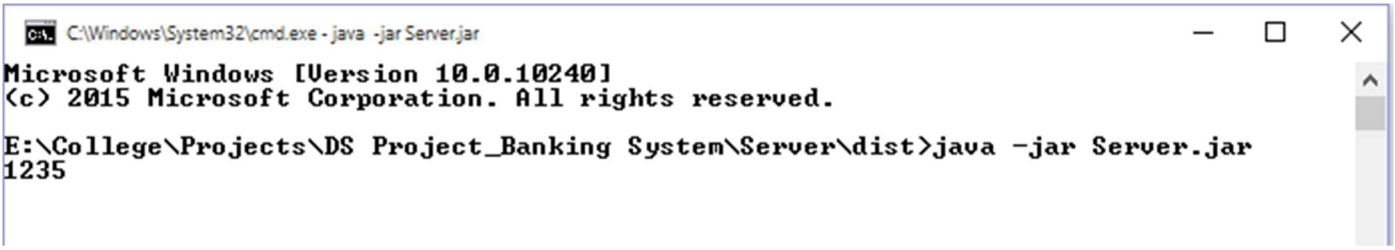

If transfer was successful,
1
Close connection
send new balance
balance

Quit

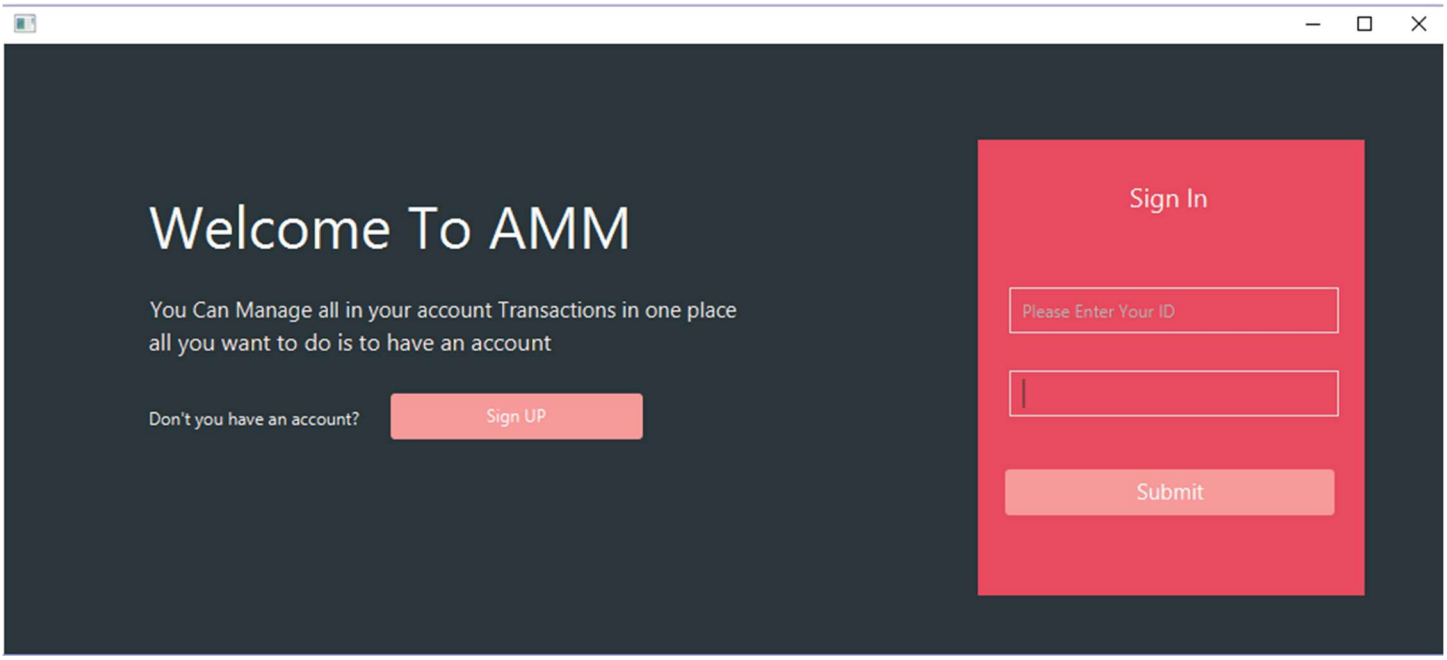

Client
Server
0
0
Client Program Ends

User Guide

1) First you need to run server program Server.jar file with Server_config.txt file next to it that contains the port number that will be used in server socket.

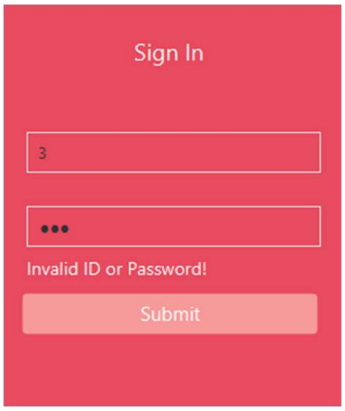


2) Run Client.jar file with Client_config.txt file next to it that contains the server IP and port number that will be used to connect to server.



3) You can sign in with your account ID and password.

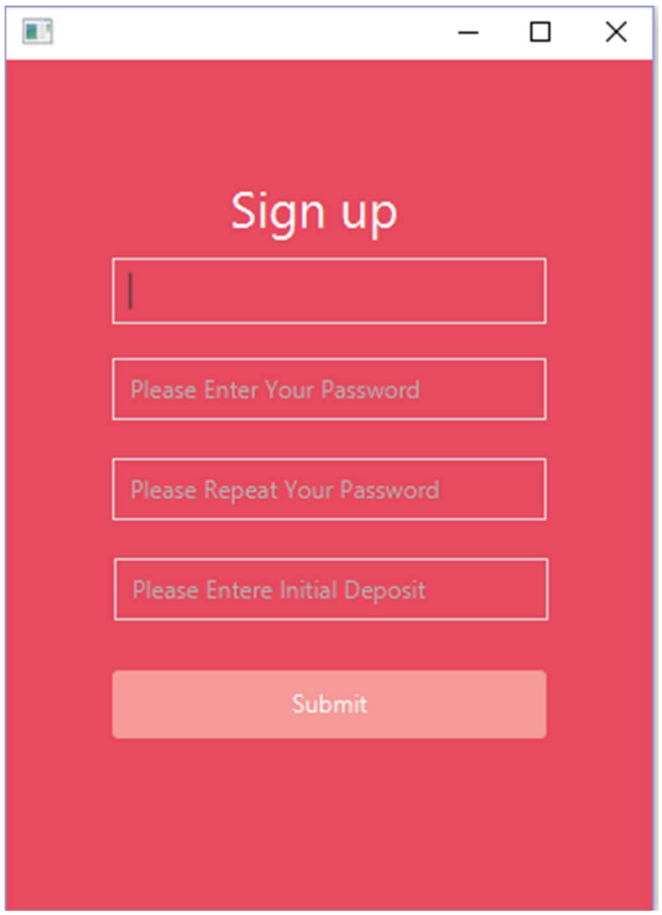
- If ID or password are wrong an error will appear.



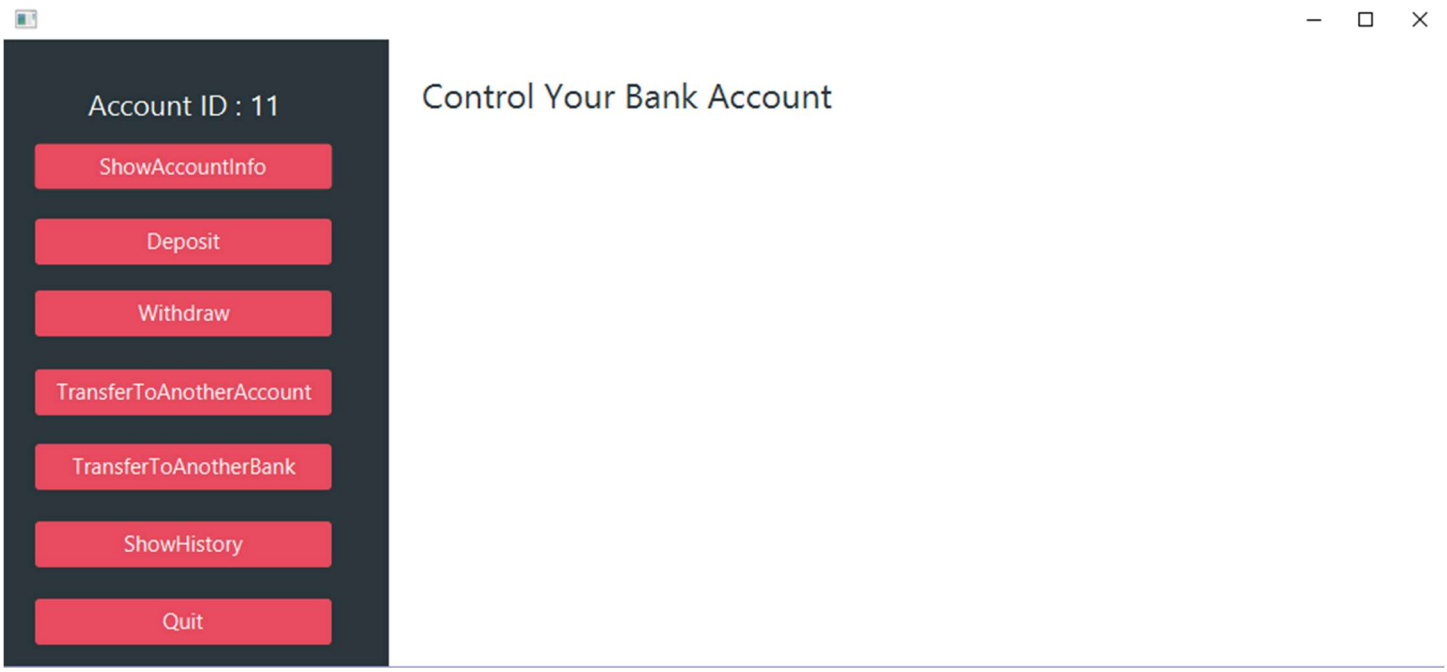
4) You can create a new account.

- you must enter all fields.

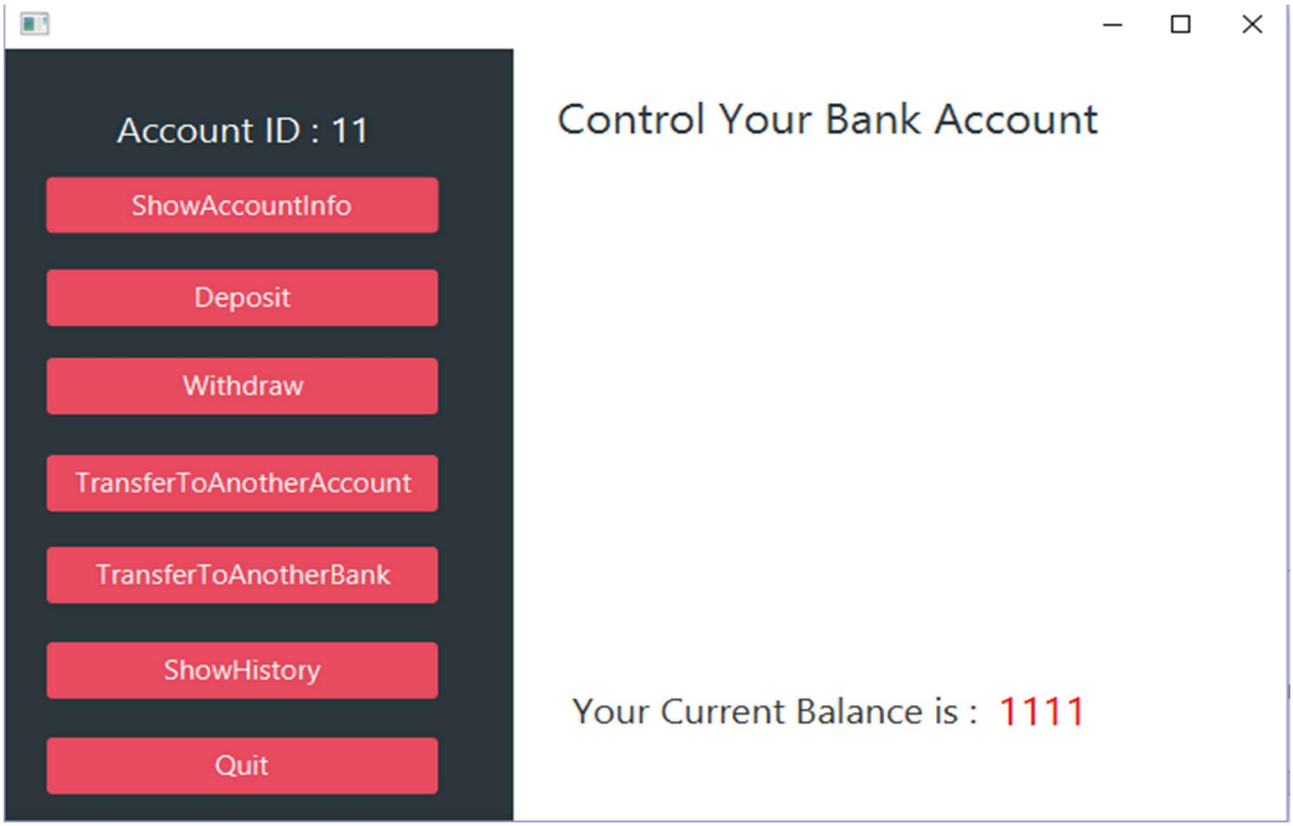
- All error cases are handled.



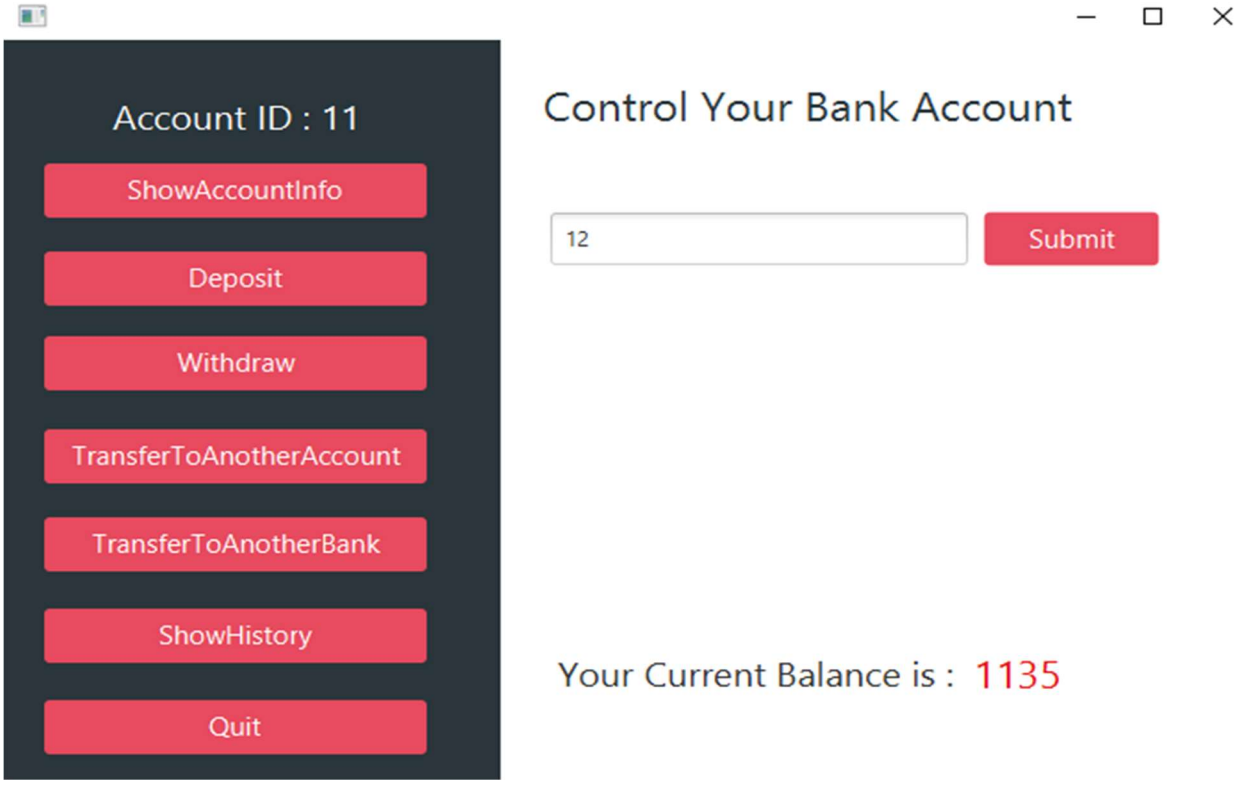
5) After Logged in, you can choose from available system services.



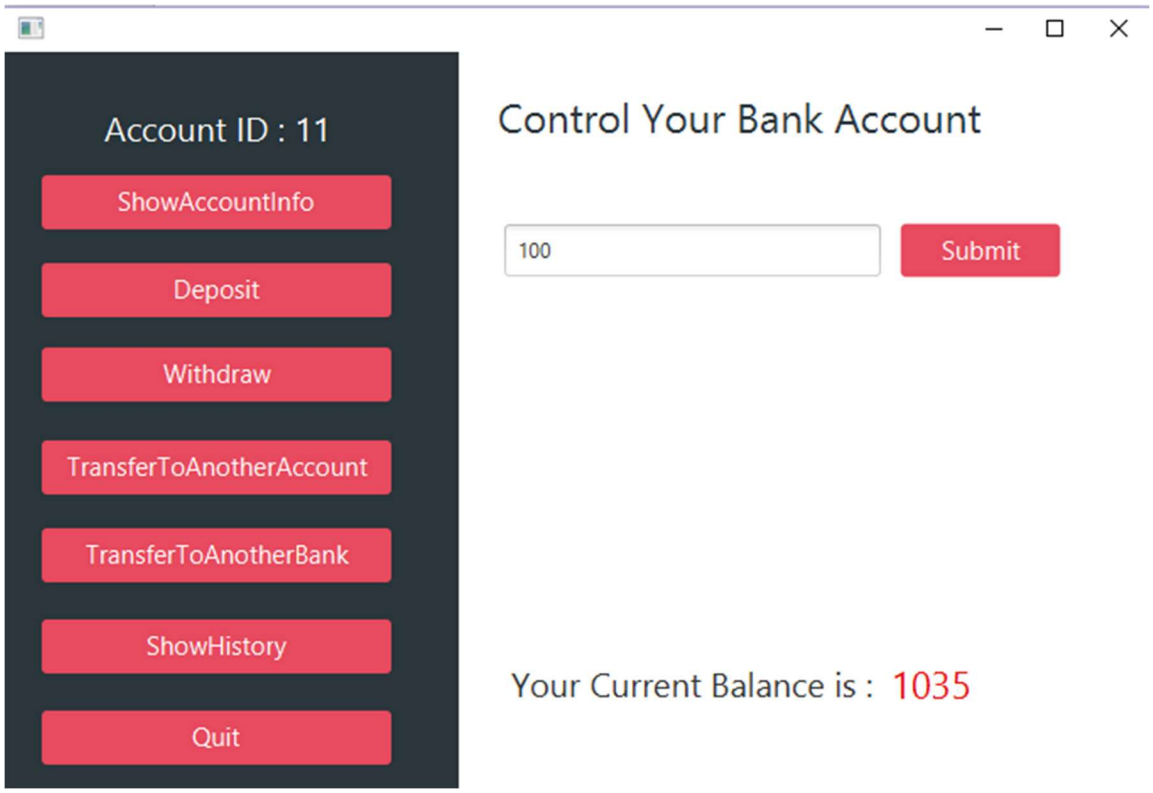
Show Account Info



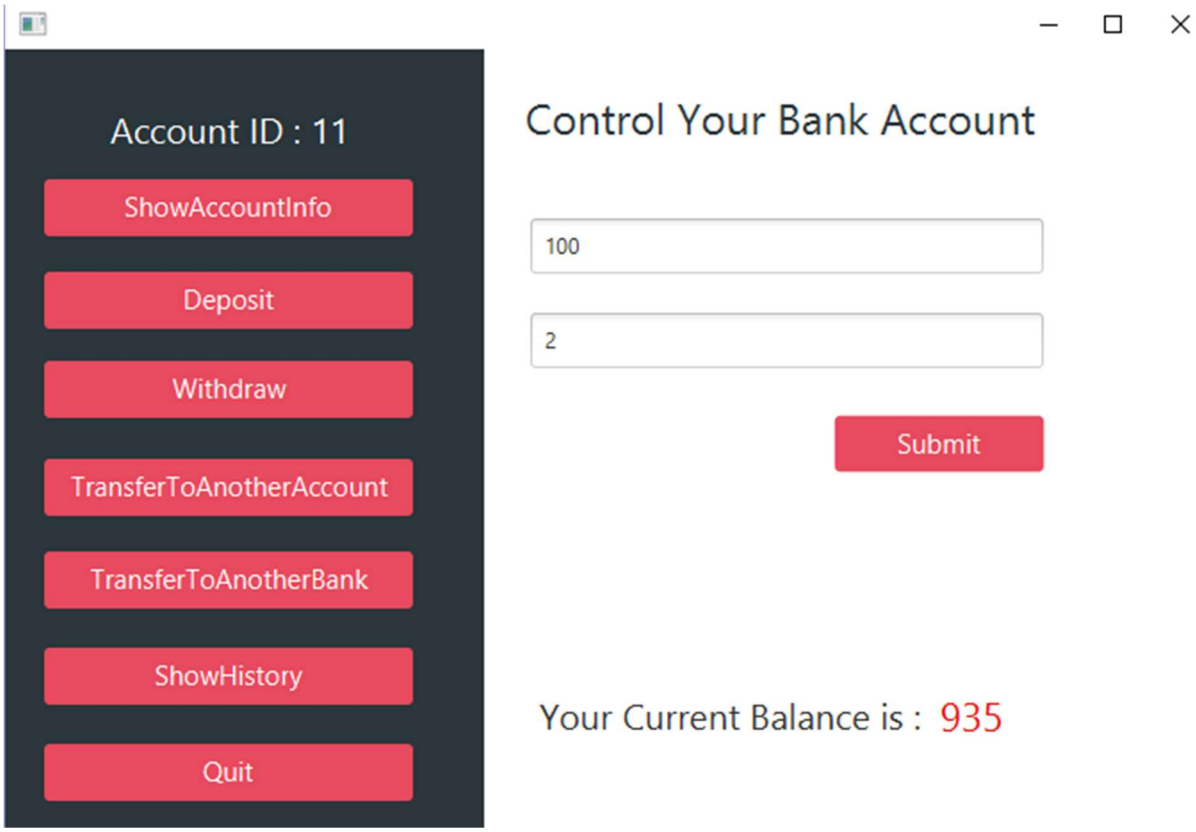
Deposit



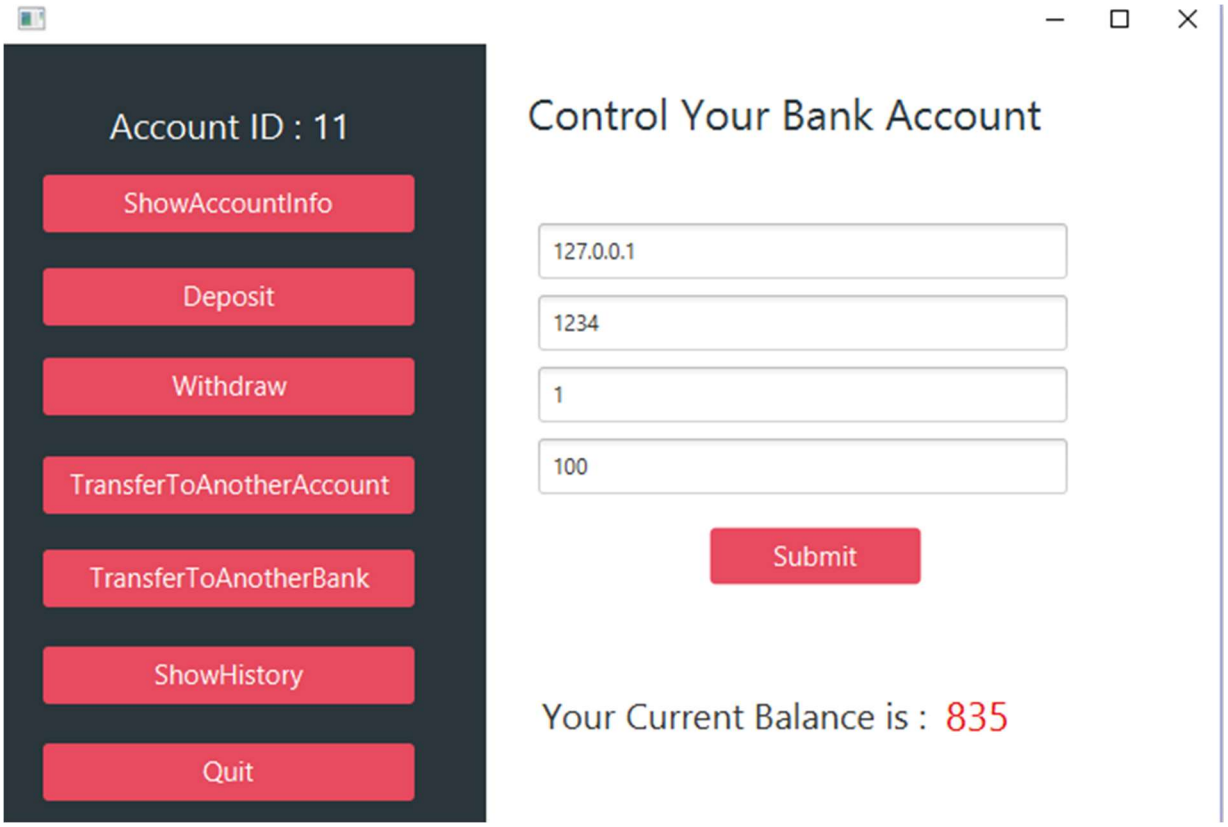
Withdraw



Transfer money to another account (in the same bank)



Transfer money to another account (another bank)



History