1. Introduction
    1.1. Goals and Objectives
        1.1.1. This documentation describes the important aspects of client and server connections between a client ATM or Teller with its banking Server.
    1.2. Statement of Scope
        1.2.1. Priorities are the maintainability, usability, portability, and efficiency of the system such that they can be easily reproduced at a different environment.
    1.3. Software Context
        1.3.1. Connections between client and server will be made through TCP connections as they are bound and connected through a given port.
    1.4. Major Constraints
        1.4.1. Keeping a constant connection between the client and server. Drop connection if the client is hanging the port to free up for other client requests to connect in.
2. Data Design
    2.1. Client Side
        2.1.1. PrintWriter will be used for socket outPutStream to send server messages in regards to the client's account (i.e. deposit, withdraw, or transfer). Messages will be multithreaded due to many clients.
        2.1.2. BufferStream will be needed to receive messages from the server after validation before sending to the UI for display.
    2.2. Server Side
        2.2.1. Server will be listening on a specific socket while reusing the address for future connections from other clients.
        2.2.2. Server socket will be listening for client's messages to validate a user account before proceeding with any transactions.
        2.2.3. ClientSock thread will be handling each client message separately
        2.2.4. getClient(msg) will be called to retrieve client's information from Customer[] customers.
        2.2.5. Information retrieved from customers will be used to process further incoming messages for impending transactions or balance checking.
3. Architecture Diagram
    3.1. Description of Client
        3.1.1. Client Processing Narrative
            3.1.1.1. ATM Client: ATM will wait for ATM user to insert ATM card into machine and will connect to specific socket on server to establish ATM Connection.
            3.1.1.2. Teller Client: Teller will send and receive messages directly from server after establishing a teller customer connection and these messages will carry instructions for an operation to be carried out.
        3.1.2. Client Interface Description
            3.1.2.1. ATM Client: A GUI that displays buttons(for transaction operations) will be displayed and the ATM user will be able to

interact with the interface allowing him/her to perform authorized operations on account. The buttons displayed will be “Deposit”, “Transfer”, “View Balance”, “Withdrawal” It will also contain Panels that will display account information and relevant text.

3.1.2.2. Teller Client: Similar GUI design to ATM Client but will display more buttons as Teller has access to more operations. GUI will include panels that include account owner’s information and teller information.

3.1.3. Client Processing Details

3.1.3.1. ATM - Server: There will be a socket in the server that will listen to ATM connections. An ATM connection will be initiated once an ATM user inserts a card into the ATM. When that happens the server socket will authorize connection which then allows the ATM User to login. While this connection is established, the ATM performs operations for the user by sending messages to the server who then accepts these messages and lets the ATM know if the operation was successful or not by passing a message. If the operation is successful then the server records whatever operation it was. When the ATM card is ejected the ATM connection to the server is closed.

3.1.3.2. Teller - Server: There will also be a socket in the server that will listen to teller customer connections. A teller connection will be initiated once a teller sends a message to server to verify login and when the server sends a success message back to the teller the connection will be established. This connection gives the teller access to several user’ accounts. The teller can send messages to the server to complete operations on behalf of an account owner and when these operations are complete the teller accepts a success message from the server which will be displayed to the teller as well as updates on the account. When a teller logs out, the connection will close.

3.2. Description of Server: The server will have sockets that are listening to any attempt at a connection from both ATM and teller clients. When a connection is established it gives the client's authority to perform operations through the server provided the rules set by the server for the operation to be carried out have been met. These clients communicate these operations by sending messages to the server. The server is responsible for analyzing the data and performing heavy tasks by itself. These tasks include, storing data, updating data, and managing connections.

3.3. Software Interface Description

3.3.1. External Interfaces

3.3.2. Internal Interfaces -

3.3.2.1. Check Validation

3.3.3. Human Interfaces

4. User Interface Design
    4.1. ATM
        4.1.1. Text: Welcome Text, Customer information text
        4.1.2. Buttons: “Deposit”, “Withdrawal”, “Transfer”, “View Balance”, “Remove Card”. Three(3) rows and two(2) columns with a huge gap between columns.
        4.1.3. Panels: Three panels, one on top, one below and one pop up panel. Top panel to display text, second panel to display Buttons and third panel to display account balance.
    4.2. Teller
        4.2.1. Text: Welcome Text, Customer information text, Teller information text
        4.2.2. Buttons: “Deposit”, “Withdrawal”, “Transfer”, “View Balance”, “Teller Sign Out”, “Customer Sign Out”, “Make Payment”, “Add Account”, “Close Account”. Five(5) rows and two(2) columns with a huge gap between columns.
        4.2.3. Panels: Three panels, one on top, one below and one pop up panel. Top panel to display text, second panel to display Buttons and third panel to display account information.
5. Restrictions, Limitations, and Constraints
    5.1. Restrictions
        5.1.1. Only the Java language can be used
        5.1.2. No Web Browser technologies
        5.1.3. Internet connections must be over TCP/IP
    5.2. Limitations
        5.2.1. Only Teller and ATM use cases
        5.2.2. No actual banking systems will be utilized (counterfeit, check validation, ATM cards, physical ATM, real currency)
    5.3. Constraints
        5.3.1. Zero funding for this project
        5.3.2. Only a staff of 3 working part time (with some consultation)
        5.3.3. Project must be completed in 6 weeks
        5.3.4. Only personally owned on-hand hardware and residential networking
6. Testing Issues (TBD)
7. Appendices N/A