CS 6650 Distributed System Final Project Proposal

Group Member: Chien-Yu Chen, Qing Peng, Zhaoyang Wang, Tingting Jiang
Project description: Chat and Charity (One to More Chatting Room with One to One Transaction System)
Project Name: Charity Now

Summary Description

This project is a platform to simplify the process of donation. Recipients of donation can create a room to promote their stories, and illustrate how they will spend the donation. People can join the rooms to chat with recipients, and donate to them if they would like to.

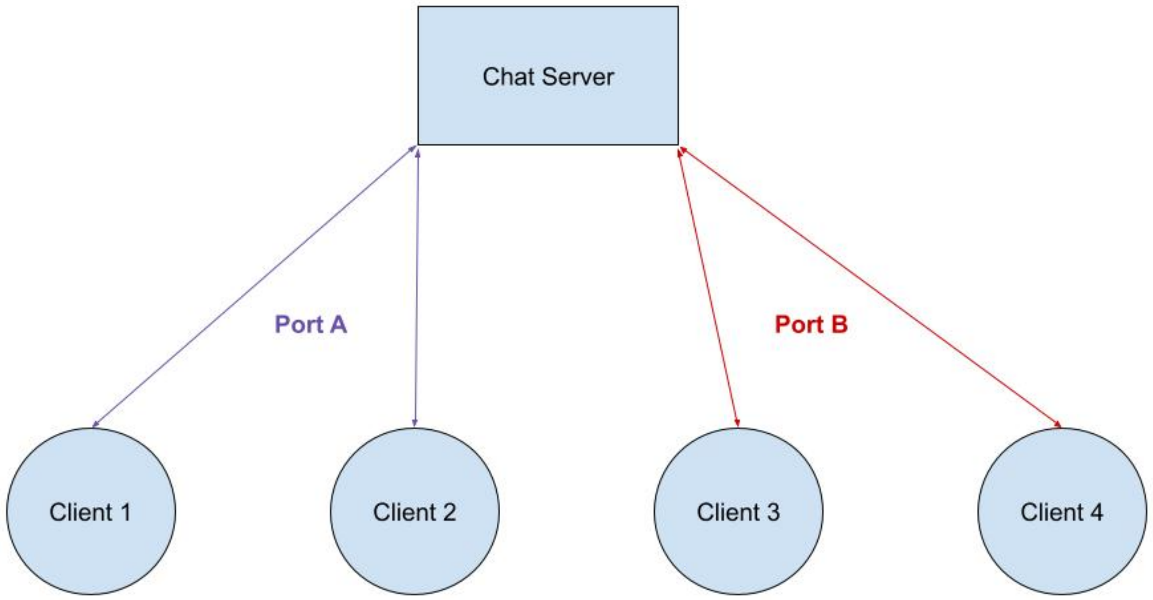
The project is a combination of Distributed Transactions, Distributed Consensus, Fault tolerance, Group Communication and Replicated data management and other significant distributed system algorithms. It builds an application to allow multiple users to join in different chatting rooms held by different users who need help from others and also enable users to make donations to the host of the chatting room.

This project is designed to realize the following two main functions:

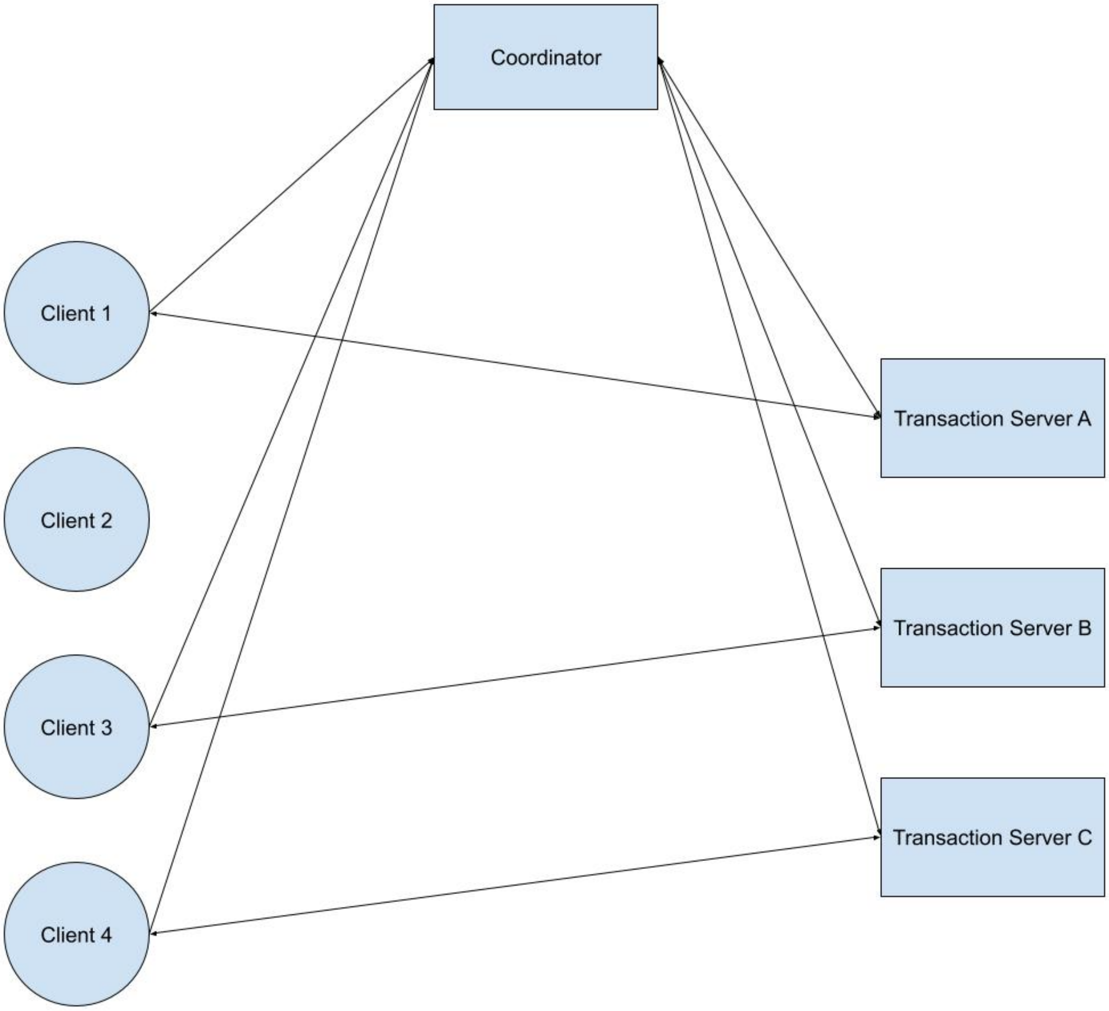
a. chatting with other users in a chatting room, it allows:
    - Private message sending
    - Real-time group chatting
    - Check the users name list
    - Change to another chatting room
b. making a one to one donation to the host of the chatting room.
    - Once a user logs in the system, the user will automatically be assigned with $1,000 in his account.
    - Check balance in his account
    - Make transactions with other users in the system

Architecture Overview Diagram(AOD) and Design Description

For the chatting system, we utilize one server with random ports for the clients' connection.



For the transaction system, each user will connect to a random server that we have setup before. By applying the Two-Phase Commit algorithm, users are able to do transactions in a distributed way and consensus.



Implementation Approach

a. In order to provide secure service, this project uses TCP and web sockets for communication.

b. To provide service located in different areas, there are several servers running in different locations. A client sends a request to the server closest to it, and if the request modifies data, this request will be synchronized to all the other servers.

c. Two-phase commit is used to ensure the atomicity of a transaction.

d. There is a User class to store all the information of a user, containing username, the socket to communicate, inputstream and outputstream. And all the instances of users are recorded into a list.
e. For each user, there is a seperate thread created to handle the requests of the user.
f. The user interface will be implemented in javafx, containing user login, chatting room and money donation interfaces.

Key Algorithms Involved

a. **Fault tolerance**: When the user is disconnected from the server, other clients should not be affected by using the chat system.
b. **Distributed Consensus**: A transaction can take place successfully if and only if the server receives acknowledgments from both clients. Otherwise, rollback operation should proceed.
c. **Group Communication**: Group chat functionality would apply the group communication algorithm.
d. **Time and Clocks**: When two clients send messages between each other, for example, client A sends a message to client B. Then, client A's system time would be recorded. When client B sends another message to client A, to calculate client B's sending time, the Lamport algorithm would be applied.

Expected Result

This project applies several key distributed system algorithms to design a real-time chatting and donating system, which supports multiple clients to access the system and connect to a random server to join the chatting room.

Function supported for Chatting system:

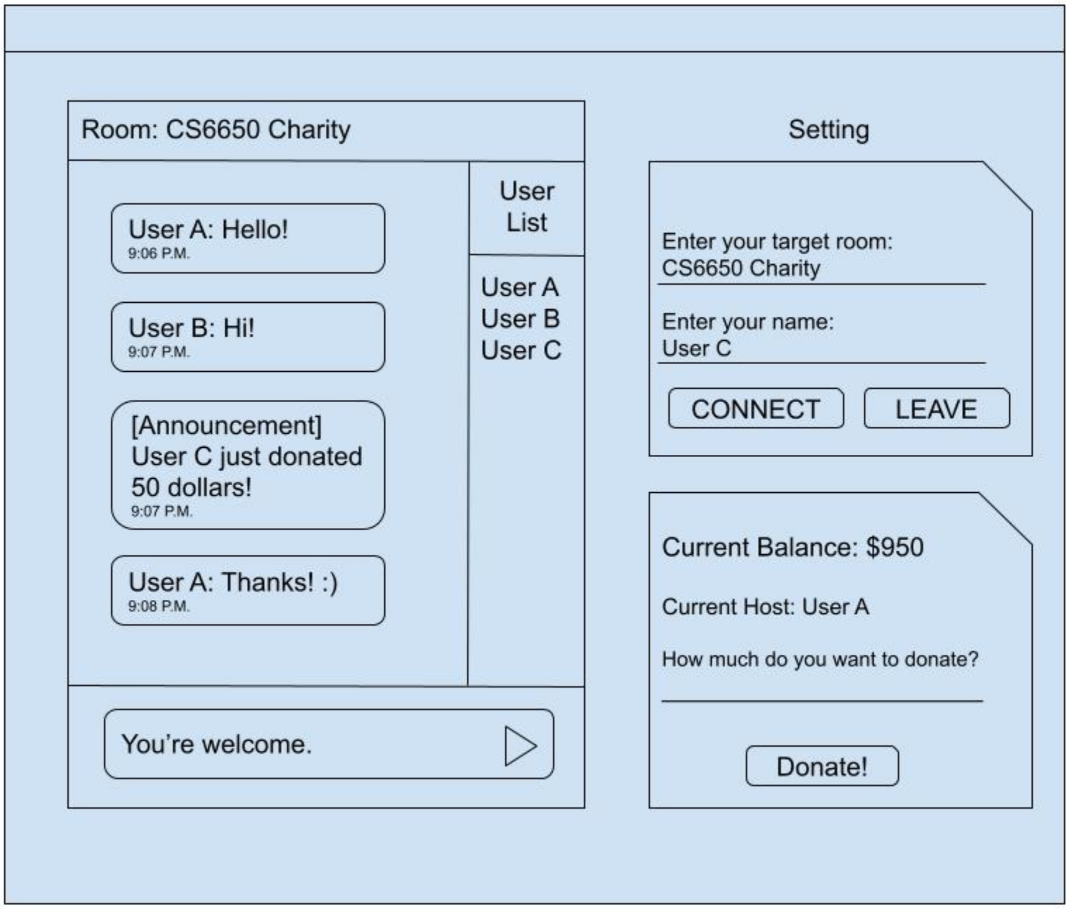
a. Group chat: Allows a maximum of 500 people to chat in the same chatting room, if a client sends a message to the group, all other members in the group should see the message.

b. Private chat: Client A can send secret messages to client B (no other people can see)
c. Display the member lists who joins in the chatting room
d. Clients can join in the designated chatting room (if room exists) properly, warning messages should pop out if the client types in the wrong command.

Function supported for Transaction system:

a. Transaction between two clients: When clients make a transaction request, if two clients satisfy the transaction condition, the transaction would be executed. If the client's balance was less than the transaction amount (sending amount) or other unexpected circumstances happened, an error message should pop out, then the transaction would be terminated.
b. Clients can check the balance in the account.
c. Clients can see the amount of money donated in the current charity lobby.

Overview:



Room: CS6650 Charity

User A: Hello!
9:06 P.M.

User B: Hi!
9:07 P.M.

[Announcement]
User C just donated 50 dollars!
9:07 P.M.

User A: Thanks! :)
9:08 P.M.

You're welcome.

Enter your target room:
CS6650 Charity

Enter your name:
User C

Current Balance: $950

Current Host: User A

How much do you want to donate?