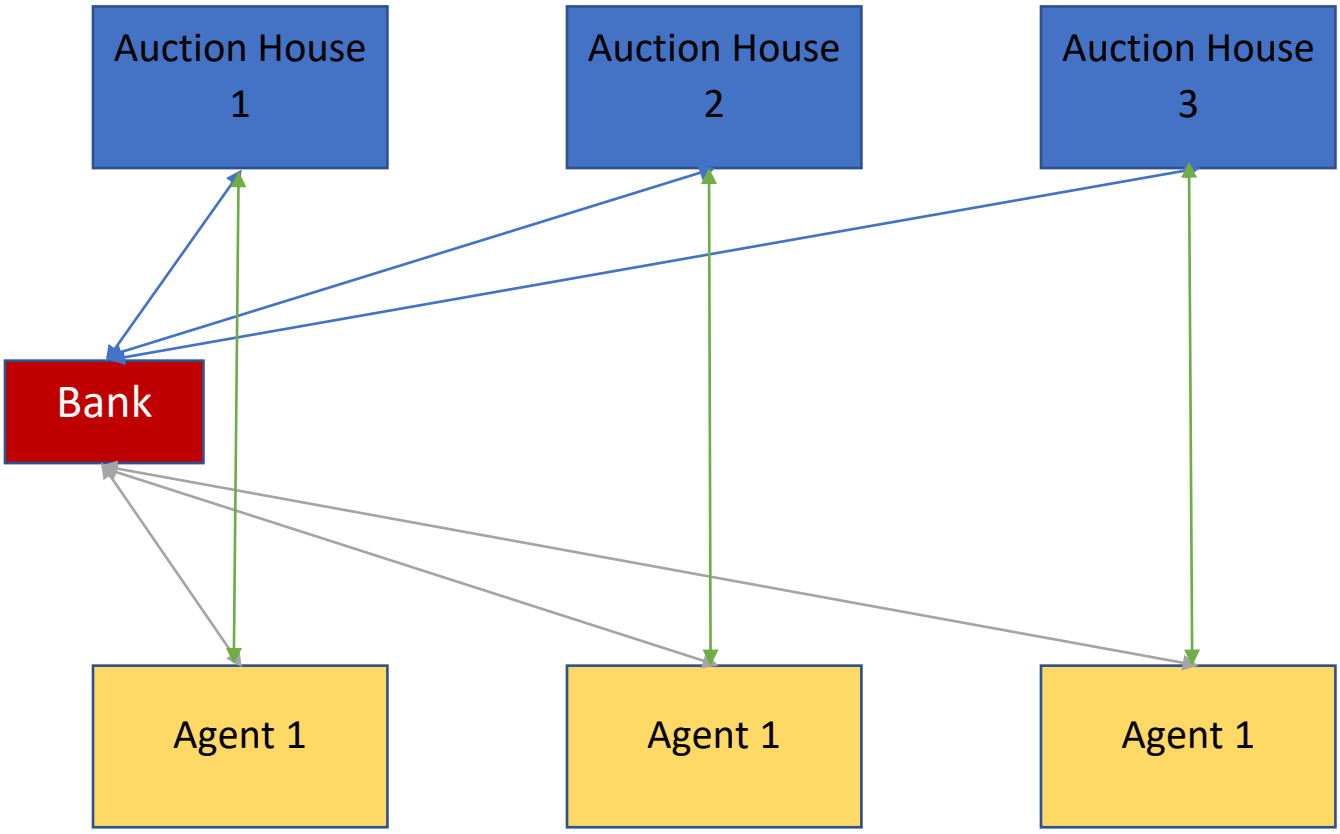
Project 5: Distributed Auctions Object Design

Team: Trey Sampson, Charley Bickel, Biraj Silwal

Proposed Design



- The Bank class is represented by the red square. This serves as a static server that every client connects to before performing any other action. The bank communicates with its clients via sockets through two classes: AgentClient and AuctionHouseClient. Each new client object the bank creates runs on its own dedicated thread so that it is capable of concurrent communications.
    - AgentClient: handles all the communication between the bank and the agent it represents as a proxy object. These are represented by the grey lines on the design
    - AuctionHouseClient: handles all communications between the bank and the auction house it represents as a proxy object. These are represented by the blue lines on the design.
- The bank is not the only class to function as a socketServer in the program. Regardless of order of connection, each agent is provided the ip addresses of each connected auction house so they may connect directly to the auction house via a client socket. These are represented by the green lines in the design.
- Auction houses accepting client connections from agents on their socketServers continuously maintain their client connection to the bank in order to send instructions based on user interaction. As agents bid on items the corresponding auction house instructs the bank to block and unblock the necessary funds.
- Every class throughout the project makes use of proxy objects running on independent threads for communication over sockets with the various servers they connect to.