

# CS351 Auction House Project - Auction House Message System Outline

Andrew Geyko

This document serves to outline the how the Auction House utilizes the messaging system in order to provide a (hopefully) accessible format for creating components that interact with it. I believe that it would be easier for the bank to specify it's own standard of how it receives messages since it seems that the AuctionHouse/Agent are requesting things for the bank to do, and specifying something like that here would just put unnecessary constraints on the bank code.

*This document is a work in progress, please let me know if some change in this format would make it easier to write your code, we can work on changing it*

## Initial Connection

An initial connection is established when a socket is first opened. When the AuctionHouse receives the following message, it will send a response acknowledging that the connection was successful. If the response is not received, something has probably broke.

*Message (Agent→AH) Format*

*Origin:* Origin.AGENT

*Type:* Type.ESTABLISH\_CONNECTION

*Body:* Bank Account ID number should be provided here

*Info:* N/A

*Message (AH→Agent) Format*

*Origin:* Origin.AUCTIONHOUSE

*Type:* Type.ACKNOWLEDGE\_CONNECTION

*Body:* N/A

*Info:* N/A

# Main Logic

## Getting Items

The get items message will be used by the agent to get a list of the items being auctioned along with their ID numbers (which will be used to specify which item to bid on).

*Message (Agents→AH) Format*

*Origin:* Origin.AGENT

*Type:* Type.GET\_ITEMS

*Body:* N/A

*Info:* N/A

*Message (AH→Agent) Format*

*Origin:* Origin.AUCTIONHOUSE

*Type:* Type.SEND\_ITEMS

*Body:* N/A

*Info:* Sends a list of objects of type “Item”

## Bidding

The bid message will be used to make a bid (clearly). It will be necessary to specify the amount that the agent is bidding, and the ID of the auction the agent wants to bid on. In response, the agent will receive a message stating whether the bid was accepted or denied.

*Message (Agents→AH) Format*

*Origin:* Origin.AGENT

*Type:* Type.MAKEBID

*Body:* This will be two separate lines (separated by newline character). The first line will specify an integer amount for the amount to bet. The second line will be an integer id of the auction to bet on (this id is sent in the “get items” message)

*Info:* N/A

*Message (AH→Agent) Format*

*Origin:* Origin.AUCTIONHOUSE

*Type:* BIDSUCCESS — BIDFAILED (self-explanatory)

*Body:* Maybe a string outlining the reason for the failure, but maybe not necessary.

*Info:* N/A

## Winning a Bid

In this case, the agent does not send a message, but just gets one whenever it wins. Although a response to to the auction house is not necessary, it is necessary to send a message to the bank in order to transfer the funds to the auction house.

*Message (AH→Agent) Format*

*Origin:* Origin.AUCTIONHOUSE

*Type:* BID.WON

*Body:* ID number of the auction house, sent to the bank in order to transfer funds (this ID is also the AH bank account id)

*Info:* Will send an Item object over

## Outbid by other agent

Upon getting outbid, the agent will receive a message letting them know they have been outbid, and the AH will take care of unblocking the funds for the agent, no response should be necessary.

*Message (AH→AGENT) Format*

*Origin:* Origin.AUCTIONHOUSE

*Type:* BID.OUTBID

*Body:* first line is auction house id, second line is auction ID

*Info:* N/A

## Closing Connection

Closing a connection is handled by the Auction House at any point in time, but in order to ensure that this is handled smoothly, the agent should double check that they have no active bid, which is what this message does. When sending the can\_close message, the auction house will check and see if the agent has any active bids and respond accordingly.

*Message (AGENT→AH) Format*

*Origin:* Origin.AGENT

*Type:* CHECK\_CLOSE

*Body:* N/A

*Info:* N/A

*Message (AH→AGENT) Format*

*Origin:* Origin.AUCTIONHOUSE

*Type:* CAN\_CLOSE or CANT\_CLOSE

*Body:* N/A

*Info:* N/A