

Messages Guide

Table of Contents

Note	2
Section 1: Client Messages	3
“ctrl” Messages	3
“cmmd” Messages	3
“chat” Messages	4
Section 2: Host Server Messages	5
“ctrl” Messages	5
“cmmd” Messages	5
“chat” Messages	5

Note

The following messages guide has been written by Botond Kreicz, one of the developers responsible for the creation of the website *blackjackorbust.000webhostapp.com* and its contents. This guide is aimed at users experienced with Python. This guide gives a detailed insight into how the client and the servers communicate and as such should not be used by inexperienced users unfamiliar with Python and/or the client-server model.

All messages are in the format of (ipAddress, messageType, message). ipAddress is always the IP address of the sender and will not be mentioned again as it is outside the infrastructure's control. messageType can be “ctrl”, “cmmd” or “chat” and each of those is discussed in the subsections of each section. message can be static, such as “start” and “stop”, or dynamic, such as the chat messages created by the users. Both message types will be discussed.

Section 1: Client Messages

“ctrl” Messages

The introductory message (in the form: (playerName, playerFunds, playerBet)), sent when the player presses the “Connect” button on his/her client GUI, tells the host server that the player wishes to join the game. The host server takes the player's statistics, saves them and adds the player to its list of players.

The “start” message, sent when the player presses the “Connect” button on his/her client GUI, tells the host server that the player wishes to start the game. The host server removes two cards from the deck, updates the player's statistics and sends the relevant statistics as well as the two cards' information to the player's game server.

The “stop” message, sent when the player presses the “Disconnect” button on his/her client GUI, tells the host server that the player wishes to leave the game. The host server removes the player's statistics and sends the player's game and chat servers a message to stop.

“cmmd” Messages

The “hit” messages, sent when the player presses the “Hit” button on his/her client GUI, tells the host server that the player requests another card. If it is the player's turn, the host server removes a card from the deck, updates the player's statistics and sends the relevant statistics as well as the card's information to the player's game server. If the player goes over 21, the host server ends the player's turn and sends a message to the player's game server.

The “stick” message, sent when the player presses the “Stick” button on his/her client GUI, tells the host server that the player requests to end his/her turn. If it is the player's turn, the host server ends the player's turn and sends a message to the player's game server.

The “double_down” message, sent when the player presses the “Double Down” button on his/her client GUI, tells the host server that the player requests to double his/her bet, to get one more card and then to end his/her turn. If it is the player's first turn, the host sever doubles the player's bet, removes the difference from his/her funds, removes a card from the deck, updates the player's statistics, sends the relevant statistics and the card's information to the player, ends the player's turn and sends a message to the player's game server. If the player goes over 21, the host server sends a message to the player's game server.

The “take_out_insurance” message, sent when the player presses the “Take Out Insurance” button on his/her client GUI, tells the host server that the player requests to take out insurance. If it is the player's first turn and the dealer's face up card is an Ace, the host server takes a side bet 50% of the main bet from the player's funds as insurance. If the dealer's second card is a card with a value of 10 (i.e. 10, Jack, Queen, King), the player gets his main bet and side bet back, otherwise he/she loses

the side bet. Either way, the host server sends a message to the player's game server.

The “surrender” message, sent when the player presses the “Surrender” button on his/her client GUI, tells the host server that the player requests to surrender. If it is the player's first turn, the host server halves the player's main bet, adding one half to his/her funds, ends the player's turn and sends a message to the player's game server.

“chat” Messages

The message, sent when the player presses the “Send” button on his/her client GUI after writing into the text field, tells the host server that the player requests to broadcast his/her message. The host server takes the message from the player and iterates through its users, sending each a copy of the message to his/her chat server, signing it as the player.

Section 2: Host Server Messages

“ctrl” Messages

The “stop” message, sent to the game and chat servers of the player that sent a “ctrl” “stop” message to the host server, tells the servers to close. The servers in question close their respective sockets and return, except if the server is the host server in which case it kills the thread taking care of the player that sent the message as opposed to closing its socket and returning (though it will do that if all the players leave).

The ending message (in the form: (playerName, playerFunds, moneyWon, moneyLost, gamesWon, gamesLost, gamesPlayer)), sent when a game session ends or when the player presses the “Surrender” button on his/her client GUI to the game server. The game server then updates the database to reflect the player's statistics.

“cmmd” Messages

The message, sent to the game server of the player that sent a “cmmd” message to the host server, gives information to the game server regarding the player's statistics and his/her cards, if appropriate. The game server displays the relevant information.

“chat” Messages

The message that the host server receives, it forwards to the chat servers of every player. The chat servers display the messages with the sender's name.

Please note that for the all messages, if the player in question is the host, then instead of sending a message to the player's game server, the host server simply prints out the message as the host server acts as a player's game server.