

# Server Protocol Documentation

ServerProtocol is a class mainly containing methods to process input given to the server. This is done by checking incoming messages for commands and triggering according responses.

We are still considering and will probably later change the server protocol to an enumerated list.

## Methods

### process

The method “process” is the primary step in our protocol for input to take. Every message is checked in a switch-statement whether it is a command and accordingly calls the corresponding method. Currently, if no command is recognized the input will be processed as a message. This will probably change later.

Parameters:

- String input                      input given by client, needed for processing.
- String name                      who gave input, needed to find in the List of players.

Our possible commands for the time being are:

- /lout                      to log out.
- /nick                      to change nickname.
- /ping                      to display the players ping.
- /dmsg                      to message someone directly.
- /list                      to list all players online.
- /lbbby                      to create a lobby.
- /join                      to join a lobby.
- /race                      to force-start a race in a lobby
- /leav                      to leave a lobby.
- /loli                      to show a **lobby list**.
- /cast                      to send all online players a message.
- /read                      to set a player to a ready state for the race.
- /unrd                      to unready a player for the race.
- /help                      to see all important commands.
- /wins                      to see the high score.
- Default                      to message within lobby.
- (/evnt                      to deal with events in game)
- (/move                      to move in game)

## Messaging methods

### broadcastMessage

The method “sendMessage” is to display the input of a player to all players connected to the server. It creates a message that will be given out from the server-side with a time-stamp and name to every player.

Parameters:

- String message                      content of the message.
- String name                          name of the person who sent the message, for info.

### sendDirectMessage

The method “sendDirectMessage” is to send a message to a specific person, like a whisper-method. Only the recipient can read the message.

Parameters:

- String name                          name of the person who sent the message, for info.
- String dmname                      name of the recipient.
- String directMessage              content of the message.

## Lobby methods

### showLobbies

The method “showLobbies” is to display the user using the command a list of all lobbies.

Parameter:

- String name                          name of the person who sent the message, needed for info.

### lobbyMessage

The method “lobbyMessage” exists to send the message to all players inside of a lobby.

Parameters:

- String name                          name of the person who sent the message, for info.
- String message                      content of the message.
- String lobbyName                   name of the Lobby in which to send the message.

### lobbyError

The method “lobbyError” is a method to send a notification within a lobby, so not everyone on the server is bothered by information of another lobby.

Parameters:

- String name                          name of the person who caused a lobbyError
- String message                      content of the message to display.

### startLobbyGame

The method “startLobbyGame” is a method to start the race in a lobby, regardless of player-stated readiness.

Parameters:

- String name                          name of the person who is member of the lobby

### closeLobby

The method “closeLobby” is called upon, when a Lobby should be closed due to no players being in there anymore.

Parameters:

- Lobby closingLobby                      Lobby which should be closed.

### createLobby

The method “createLobby” is called when a player wants to create a lobby with a certain name.

Parameters:

- String lobbyName                      name of the lobby that is created.
- String creator                      name of the creator of the lobby for information.

### joinLobby

The method “joinLobby” is used, when a player wants to enter an existing lobby.

Parameters:

- String name                      name of the person wanting to join a lobby, for info.
- String lobbyName                      name of the lobby the person wants to join.

### leaveLobby

The method “leaveLobby” exists, so a player can a lobby they are currently in.

Parameters:

- String name                      name of the person wanting to leave a lobby, for info.

### setLobbyList

The method “setLobbyList” initiates a list of all players in a lobby and logs it.

Parameters:

- List<Lobby> list                      to initialize the correct list.

## General methods

### sendNotification

The method “sendNotification” is very similar to sendMessage. Instead of sending a message it gives general information about server/players in the chat.

Parameters:

- String notification                      the information displayed.
- List<Connection> players              list of all players, needed for information distribution.
- String name                              name of person who prompted the notification.

### sendMove

The method “sendMove” exists to send a move made by a player to the rest of the lobby.

Parameters:

- Move move                              to see what move has been made.

### getTimeMinutes

This method returns a String that describes the current time. The method name indicates to which level of precision this is done. The method getTimeMinutes is mostly used for chat.

### checkName

The method “checkName” checks, whether a name is already in use and if so, attaches a suffix in a consistent manner. Because on method call the name is declared by the client, it starts the counter at -1. After that, the counter will be reset to 0 when searching through the list.

Parameters:

- String name                              needs the new name to check if it already exists.

### changeUsername

The method “changeUsername” is called when a client wants to change their username by typing “/nick <newname>”. It initially calls the method checkName to check whether a username is already taken. After that, a message is displayed to inform every connected client of the event.

Parameters:

- String name                              old name to find player for whom to change their name.
- String newname                          the new name the client should be changed to.

### logout

The method “logout” is called when a client gives the command “/lout”. It gives the possibility for client to log out in a controlled manner. Now the server can disconnect the client. The method also removes the player/client from the list of all players.

Parameters:

- int playerIndex                          for logging out the correct player.

## displayPing

The method “displayPing” provides a possibility for the user to see his ping. The according command is “/ping”.

Parameters:

- String name name of the client the ping is displayed to.

## getAllPlayers

The method “getAllPlayers” gives the user a list of all Players on the server.

Parameters:

- String name name of the player who issued the command, for info.

## setPlayerList

The method “setPlayerList” initiates a list of all players on the server and logs it.

Parameters:

- List<Connection> list to initialize the correct list.

## showWins

The method “showWins” prints out the highscore of the person using the command.