

Eryantis Protocol Documentation

Legramandi, Leone, Locatelli
Group 60

May 7, 2022

1 Messages

1.1 Notification

This message is sent from the server to the client to print in client cli string messages.

Arguments

1. Notification: the string to print in client cli.

Possible responses

No response message.

1.2 Ping

This message is sent from the server to the client and vice versa to ensure that the connection is stable and there aren't any connection error.

Arguments

No Arguments.

Possible responses

Response with a Ping message.

1.3 ChooseInt

This message is sent from the client to the server, it stores a client int choice (essentially an integer).

Arguments

1. Data: the integer to be sent to the server.

Possible responses

No response.

1.4 ChooseString

This message is sent from the client to the server, it stores a client String choice (essentially a String).

Arguments

1. Data: the string to be sent to the server.

Possible responses

No response message.

1.5 ChooseAssistantCard

This message is sent from the server to the client to allows the client cli to display that the client must choose an assistant card to play the turn.

Arguments

No Arguments.

Possible responses

1. ChooseInt: message sent to the Server with the assistant card value chosen.

1.6 ChooseCloudTile

This message is sent from the server to the client to allows the client cli to display that the client must choose a Cloud Tile to get new students for the entrance room.

Arguments

No Arguments.

Possible responses

1. ChooseInt: message sent to the Server with the cloud tile index chosen.

1.7 ChooseIsland

This message is sent from the server to the client to allows the client cli to display that the client must choose an island to perform an action (e.g. move a student/mother nature on it)

Arguments

No Arguments.

Possible responses

1. ChooseInt: message sent to the Server with the index of the island chosen.

1.8 ChooseNickname

This message is sent from the server to the client to allows the client cli to display that the client must choose a nickname for the game.

Arguments

No Arguments.

Possible responses

1. ChooseString: message sent to the Server with the nickname chosen.

1.9 ChooseLobbySize

This message is sent from the server to the client to allows the client cli to display that the client must choose a lobby size (number of player in the game).

Arguments

No Arguments.

Possible responses

1. ChooseInt: message sent to the Server with the lobby size chosen between 2 or 3.

1.10 ChooseStudentColourToMove

This message is sent from the server to the client to allows the client cli to display that the client must choose a student to move.

Arguments

No Arguments.

Possible responses

1. ChooseString: message sent to the Server with colour of the student to move.

1.11 ChooseWhereToMove

This message is sent from the server to the client to allows the client cli to display that the client must choose if it wants to move a student to the dining room or to an island.

Arguments

No Arguments.

Possible responses

1. ChooseString: message sent to the Server with "Dining Room" or "Island"

1.12 UpdatePlayGround

This message is sent from the server to the client to allows the client model to be updated.

Arguments

1. PlayGroundUpdated: this is an object that contains the updated playground with all info of the game.

Possible responses

No response.

1.13 Winner

This message is sent from the server to the client to allows the client to know that there is a game winner and to know who is that winner.

Arguments

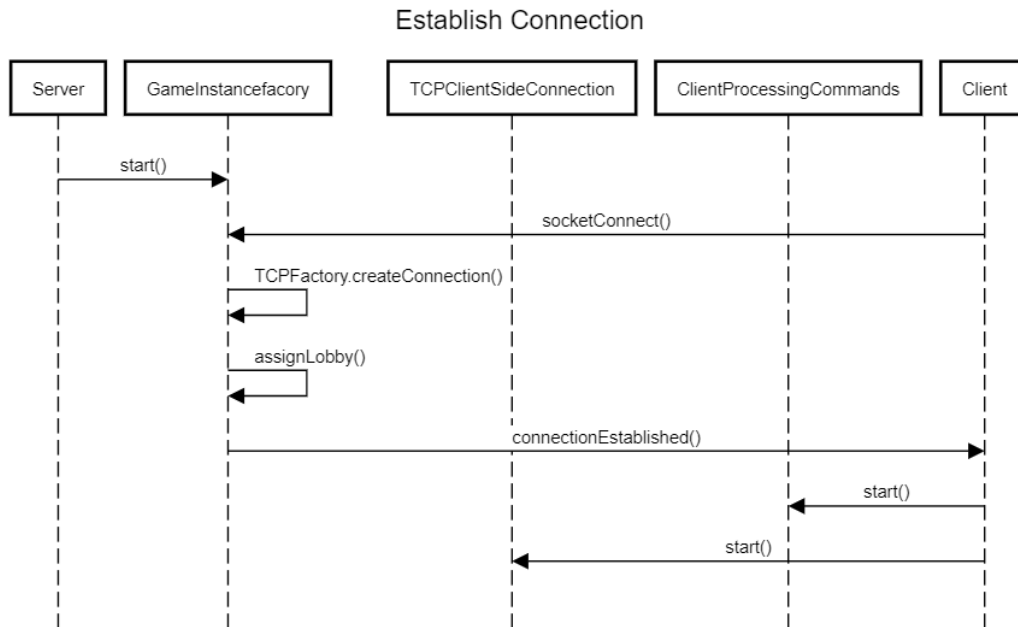
1. Winner: the nickname of the game winner

Possible responses

No response.

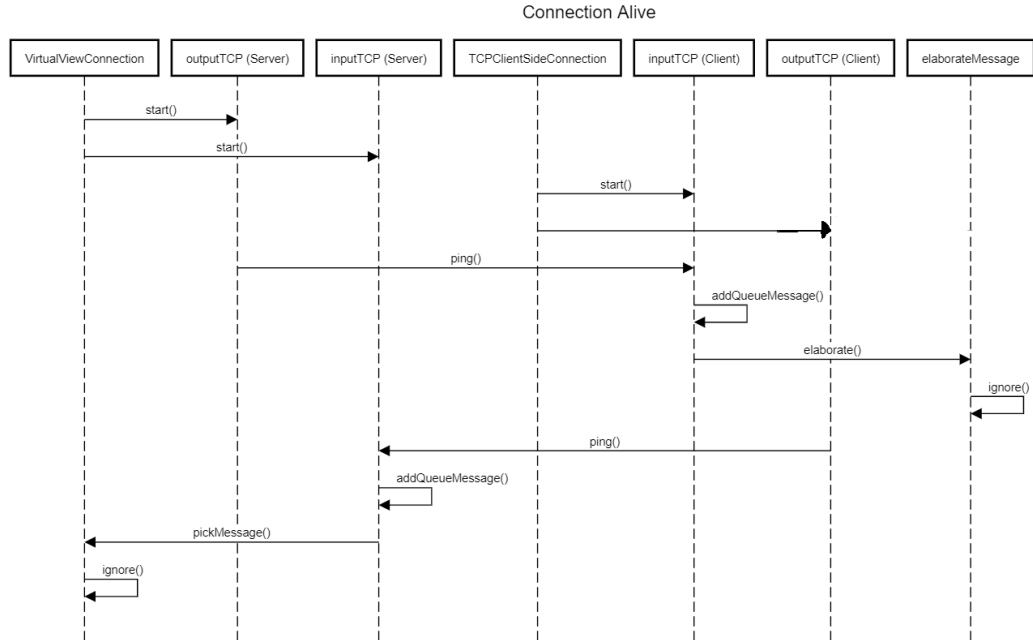
2 Scenarios

2.1 Establishing Connection



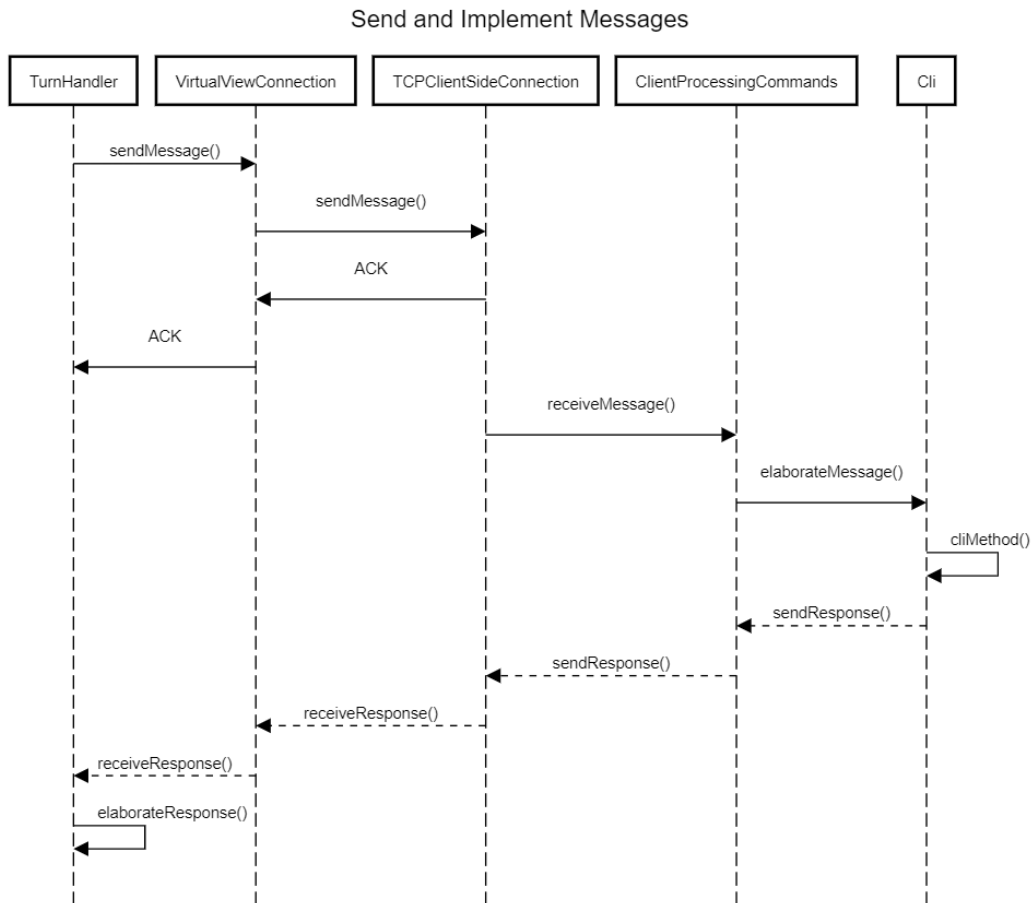
First connection of the Client to the server. The server start a games factory thread that allows to listen for new socket connection, create a new tcp connection instance and responds to client requests. The Client send a socket tcp connection request to the server, the server create a tcp connection (thanks to the factory associated) and assign the connection to a new lobby. Then, the server notify the Client that it accept the connection. The Client start threads for tcp client connection, that allows to create a stable tcp connection with the server, and for client processing commands that allows to elaborate and implement the messages received from the server.

2.2 Detect Disconnection



This scenarios represent an action done in background during all the game. The virtual client connection on the server starts two threads out/in tcp for continuously send and receive ping message. In client side part there is the class tcp client side connection that create the two threads out/in tcp for the client. All ping messages are ignored. At a specific rate of few milliseconds the ping messages are sent from the server and the client.

2.3 Send and Implement Messages



This is the way of respond to messages sent from the Server to the Client. In server side there is the "TurnHandler": the main thread that handle all action during the game. Turnhandler use the virtual connection specific of the client to send a message. The tcp client side connection handle this message and allows the client processing commands to receive and elaborate the message content by calling the right method in cli. Then the cli elaborate the message respond and send it through all previous components to the server. (Note: ACK messages are essentially the ping message explained in scenarios 2)