Server protocol:

For every request:

Check to see what type of request it is

(5 types: Create Game, token placement, Get Game, WaitResponse or TerminateGame):

If (Create Game)

If Game() object for the 2 players specified in the request already exists

Return “Welcome back to game with {the other player}” AND GameId

Else If Game() object for the 2 players specified in the request does NOT exists

Create a Game() object with if game does not already exist

Append the Game() object to the games list

Return “New game with players {the other player…}” AND GameId

Else if (Token placement)

Check the status of the Game object associated with the 2 players specified:

If Game() object doesn’t exist for the 2 players specified:

Return “Invalid game”

If Game exists:

Place player’s Token

Return “RPS placed” Game() object

Else if (Get Game):

If Game() exists:

Return Game()

Else:

Return “Game doesn’t exist”

Else if (TerminateGame):

If Game() exists:

Set Game.terminate\_{player} = True

If Game.terminate1 && Game.terminate2:

Delete Game, pop from the list

Return “Game is terminated.”

Return “Terminate Request placed”

Else:

Return “Game doesn’t exist”

Else if (WaitResponse):

While (PlayerResponse is None):

Return player response

Client Protocol:

First, make a Create Game request (Should expect a GameId being passed back with a message)

Make a Get Game request (Should expect a Game Object being passed back):

If the player has NOT placed a token:

Ask user to place token.

Make a TokenPlacement request

If the other player already placed a token:

Show Result

Else:

Send a WaitResponse request (then wait)

When result comes back:

Show Result

If the player has already placed a token:

Send a WaitResponse request (then wait)

When result comes back:

Show Result