

ClientFunctionalities	
ping()	void
playerSelection(List<String>)	String
cellSelection(List<Point>)	Point
roomSelection(List<List<Point>>)	List<Point>
reloadSelection(List<String>)	WeaponSelection
shootSelection(List<String>)	WeaponSelection
weaponSelection(List<String>)	WeaponSelection
powerupSelection(List<String>)	String
actionSelection()	TurnAction
showLeaderboard(List<String>)	void
weaponFreeSelection(List<String>)	WeaponSelection
updateMatch(JSONObject)	void
alert(String)	void

ClientPlayer	
RMI_PORT	int
SOCKET_PORT	int
server	ServerConnection
lastUpdater	Thread
loaded	boolean
ClientPlayer(String, ConnectionType, String, int, boolean)	
ClientPlayer(String)	
ping()	void
playerSelection(List<String>)	String
cellSelection(List<Point>)	Point
roomSelection(List<List<Point>>)	List<Point>
reloadSelection(List<String>)	WeaponSelection
shootSelection(List<String>)	WeaponSelection
weaponSelection(List<String>)	WeaponSelection
powerupSelection(List<String>)	String
actionSelection()	TurnAction
updateLobby()	void
showLeaderboard(List<String>)	void
weaponFreeSelection(List<String>)	WeaponSelection
updateMatch(JSONObject)	void
alert(String)	void
createGame(int, int, int, int)	void
joinGame(int)	void
backToLobby()	void
main(String[])	void
match	AdrenalinaMatch
ready	boolean
thisPlayer	Player
view	ClientView
nickname	String

«create»

ConnectionType
RMI
SOCKET

ServerConnection	
player	ClientPlayer
ServerConnection(ClientPlayer)	
connect(String, int)	void
disconnect()	void
createGame(int, int, int, int)	void
joinGame(String, int)	void
updateLobby(String)	String
backToLobby()	void
ready	boolean

RemoteClient	
server	ServerFunctionalities
PING_SECONDS	int
registry	Registry
port	int
ip	String
RemoteClient(ClientPlayer)	
connect(String, int)	void
disconnect()	void
createGame(int, int, int, int)	void
joinGame(String, int)	void
updateLobby(String)	String
backToLobby()	void
checkConnection()	void
ready	boolean

SocketClient	
socket	Socket
output	PrintWriter
ip	String
port	int
input	BufferedReader
response	String
validResponse	boolean
lock	Object
SocketClient(ClientPlayer)	
connect(String, int)	void
getResponse()	String
disconnect()	void
createGame(int, int, int, int)	void
joinGame(String, int)	void
updateLobby(String)	String
backToLobby()	void
listen()	void
sendAnswer(JSONObject)	void
sendAnswer(String)	void
parseMessage(JSONObject)	void
parseCoordinates(JSONArray)	List<Point>
lostConnection()	void
ready	boolean