Endpoint	Sample request	Sample response (200 OK)	Sample response (400+ not OK)	Notes
/rover	POST {"diagnostics": {"battery":100,"connection":100}, "MAC":1234567, "nickname":"MiWhip", "timestamp":10000, "position":[0,0], "whereat":1, "orientation":90, "branches":[20, 170, 310], "beaconangles":[100,300,250], "tofleft":10, "tofright":10, "battery":100}	{"next_actions":[0, 1, 2, 220, 3, 4, 260, 180, 5], "clear_queue":false} All possible actions are given in sample, realistically only 1 or 2 given at a time. Step – 0, spin – 1, angle – 2, idle – 3, update position – 4, done – 5. Clear queue is used in emergency stop – rover completes carrying out last action and stops, cannot restart rover.	{"error":"Incorrectly formatted request: missing MAC"}	ESP to server communication only. For whereat, 0 is passage, 1 is junction, 2 is dead end, 3 is exit. branches & beaconangles are array of identified angles. All angles in degrees from "north" clockwise. When a rover first connects, it is automatically paused, and must be manually started.
/client/allrovers	GET ~	[{"MAC":1234567, "nickname":"MiWhip", "connected":true, "sessionid":12}, {"MAC":1111111, "nickname":"Lightning McQueen", "connected":false}]	~	Returns list of all rovers. If a rover is connected, its sessionid will be given. If a rover is in the database but not connected, no sessionid will be given.

/client/replay	GET
	{"sessionid":23}
		{
		1000:[0, 0, 0, 90, 30, 30],
		2000:[100, 0, 0, 90, 35, 30],
		timestamp:[xpos, ypos, whereat, orientation, tofleft, tofright]
		}
			{"error":"Incorrectly formatted request: missing sessionid"} OR
			{"error":"Session does not exist"}
				Returns all information needed to make a replay on the client side

/client/sessions	GET ~	[{"MAC":1234567, "sessionid":23, "nickname":"Lightning McQueen"}]	~	Returns all session ids, corresponding MAC addresses and nicknames
/client/diagnostics	GET {"sessionid":"10"}	[{"MAC":"1234567", "timestamp":10000, "battery":100, "connection":100}]	{"error":"Incorrectly formatted request: missing/invalid MAC"}	Returns all logged diagnostic data for given session
/client/pause	POST {"MAC":"1234567"}	{"success":"successfully paused rover"}	{"error":"Selected rover does not exist, or is not currently connected"} OR {"error":"Incorrectly formatted request: missing MAC"}	Stops a rover from continuing its way through the maze. Idempotent.
/client/play	POST {"MAC":"1234567"}	{"success":"successfully played rover"}	{"error":"Selected rover does not exist, or is not currently connected"} OR {"error":"Incorrectly formatted request: missing MAC"}	Allows a paused rover to continue making its way through the maze. Idempotent.
/client/sessionnickname	POST {"sessionid":23, "sessionNick":"wr pace"}	{"success":"successfully changed session nickname"}	{"error":"Incorrectly formatted request: missing sessionid or sessionNick"}	Changes session nickname.
/client/rovernickname	POST {"MAC":1234567, "nickname":"YourWhip"}	{"success":"successfully changed rover nickname"}	{"error":"Incorrectly formatted request: missing MAC or nickname"}	Changes rover nickname

/client/shortestpath	POST
	{"MAC":1234567, "start":[0,0]}
		{
		1:{mapsto: -1, xcoord: 0, ycoord: 0},
		2:{mapsto: 1, xcoord: 100,ycoord: 100},
		3:{mapsto:3}, xcoord=50, ycoord=50
		}
			{"error":"Incorrectly formatted request: missing MAC or start"} OR
			{"error":"Incorrectly formatted request: invalid MAC address"}
				Returns shortest path predecessor graph, ie key is current node, mapsto is next node to travel to to ensure shortest path. Start node has mapsto -1.

/client/estop	POST {"MAC":1234567}	{"success":"estopped"}	{"error":"Missing MAC address"} OR {"error":"Invalid MAC address"}	Emergency stops given rover. Cannot be restarted, must be disconnected and reconnected.

/led_driver	POST {"current":103, "voltage":1.8}	{"success":"received data", "switch":1}	~	Controls switching beacons on/off depending on whether they are being used for a spin or not.