

Statuscode-Intervalle:

general success: 0-50

“lobby” success: 100-150

“play” success: 200-250

“debug” success: 300-350

general error: 50-100

“lobby” error: 150-200

“play” error: 250-300

“debug” error: 350-400

Seiten:

“lobby” commands : S. 1-5

“play” commands: S. 6-19

“debug” commands: S. 20

“exit” command: S.19

command	command_key	Parameter	Rückgabewert	Beschreibung	Beispiel-Request
–	–	“command”: <ul style="list-style-type: none"> “<undefined>” “command_key”: <ul style="list-style-type: none"> “<undefined>” 	Error: "response_code": 50, "response_msg": "Command <command> not found!"	Falls ein Command-Key außerhalb der definierten Command-Keys übergeben wird, Error	{ "command": "xy", "command_key": "xy" }
–	–	“command”: <ul style="list-style-type: none"> “<defined>” “command_key”: <ul style="list-style-type: none"> “<defined>” 	Error: "response_code": 51, "response_msg": "Received payload is not correct JSON!"	Falls invalides JSON übergeben wird	{ "command"= "xy", "command_key"= "xy" }

command	command_key	Parameter	Rückgabewert	Beschreibung	Beispiel-Request
„lobby“	„create“	“command”: <ul style="list-style-type: none"> “lobby” “command_key”: <ul style="list-style-type: none"> “create” 	Success: "response_code": 100, "response_msg": "Lobby created!", "key": "<lobby-key>"	Kreiert Lobby im Server	{ “command”: “lobby”, “command_key”: “create” }
			Error: "response_code": 150, "response_msg": "Client already in a lobby!"		

command	command_key	Parameter	Rückgabewert	Beschreibung	Beispiel-Request
„lobby“	„join“	“command”: <ul style="list-style-type: none"> “lobby” “command_key”: <ul style="list-style-type: none"> “join” “key”: <ul style="list-style-type: none"> “<lobby-key>” “pos”: <ul style="list-style-type: none"> “p1” “p2” “sp” 	<div> Success: <pre> "response_code": 101, "response_msg": "Joined Lobby!", "key": "<lobby-key>" "pos": "<pos>" </pre> </div> <div> Error: <pre> "response_code": 151, "response_msg": "Lobby does not exist!" "key": <lobby-key> "response_code": 150, "response_msg": "Client already in a lobby!" "response_code": 152, "response_msg": "Failed to join lobby! Maybe position blocked?" "key": <lobby-key>" </pre> </div>	Der Lobby beitreten (zweiter Spieler, Spectator)	{“command”: “lobby”, “command_key”: “join”, “key”: “<lobby-key>”}

command	command_key	Parameter	Rückgabewert	Beschreibung	Beispiel-Request
„lobby“	„leave“	“command”: <ul style="list-style-type: none"> “lobby” “command_key”: <ul style="list-style-type: none"> “leave” 	Success: “response_code”: 102, “response_msg”: “Client left lobby.”	Die Lobby verlassen	{“command”: “lobby”, “command_key”: “leave”}
			Error: “response_code”: 153, “response_msg”: “Client not in a lobby!”		
„lobby“	„swap“	“command”: <ul style="list-style-type: none"> “lobby” “command_key”: <ul style="list-style-type: none"> “swap” “pos”: <ul style="list-style-type: none"> “p1” “p2” “sp” 	Success: “response_code”: 103, “response_msg”: “Client swapped!” “pos”: “<pos>”	Die Position in der Lobby wechseln. p1 ist jener Spieler, der den ersten Zug macht, heißt will man tauschen, wer den ersten Zug machen soll, swap benutzen	{“command”: “lobby”, “command_key”: “swap”, “pos”: “p2”}
			Error: “response_code”: 154, “response_msg”: “Position unknown!” “pos”: “<pos>” “response_code”: 153, “response_msg”: “Client not in a lobby!” “response_code”: 155, “response_msg”: “Position already occupied!” “pos”: “<pos>”		

command	command_key	Parameter	Rückgabewert	Beschreibung	Beispiel-Request
„lobby“	„pos“	“command”: <ul style="list-style-type: none"> “lobby” “command_key”: <ul style="list-style-type: none"> “pos” 	Success: "response_code": 104, "response_msg": "Client position is:" "pos": "<pos>"	Gibt die Position eines Clients zurück (p1, p2 oder sp (spectator))	{“command”: “lobby”, “command_key”: “pos”}
			Error: "response_code": 153, "response_msg": "Client not in a lobby!"		
„lobby“	„status“	“command”: <ul style="list-style-type: none"> “lobby” “command_key”: <ul style="list-style-type: none"> “status” 	Success: "response_code": 105, "response_msg": "Status of lobby:", "P1": "True", "P2": "False", "Spectators": 0, "GameClient": "True", "key": "<lobby-key"	Gibt den Status einer Lobby zurück	{“command”: “lobby”, “command_key”: “status”}
			Error: "response_code": 153, "response_msg": "Client not in a lobby!"		

command	command_key	Parameter	Rückgabewert	Beschreibung	Beispiel-Request
„lobby“	–	“command”: <ul style="list-style-type: none"> “lobby” “command_key”: <ul style="list-style-type: none"> “<undefined>” 	Error: "response_code": 50, "response_msg": "Command <command_key> not found!"	Falls ein Command-Key außerhalb der definierten Command-Keys übergeben wird, Error	{"command": "lobby", "command_key": "xy"}

command	command_key	Parameter	Rückgabewert	Beschreibung	Beispiel-Request
„play“	—	“command”: <ul style="list-style-type: none"> “play” “command_key”: <ul style="list-style-type: none"> “<defined>” 	Error: “response_code”: 250, “response_msg”: “No game client connected. Try again later!” “response_code”: 251, “response_msg”: “A spectator can not play!” “response_code”: 153, “response_msg”: “Client not in a lobby!”	Valider Command und Command-Key, aber sonstige Error, die auftreten können in Verbindung mit Command „play“	/
		“command”: <ul style="list-style-type: none"> “play” “command_key”: <ul style="list-style-type: none"> “<undefined>” 	Error: “response_code”: 50, “response_msg”: “Command <command_key> not found!”	Falls ein Command-Key außerhalb der definierten Command-Keys übergeben wird, Error	{“command”: “play”, “command_key”: “xy”}

command	command_key	Parameter	Rückgabewert	Beschreibung	Beispiel-Request
/	/	/	<p>Success:</p> <p>"response_code": 202, "response_msg": "Game over:", "result": 1, "turn": 11 [broadcast, einmalig]</p> <p>"response_code": 205, "board": "[0,1,0,0]" [spielerspezifisch, konstant]</p> <p>"response_code": 206, "representation": <konsolenausgabe_board> [broadcast, konstant]</p> <p>Image <image> [broadcast, konstant]</p>	<p>Wenn im "game loop" seiend, wird initial beim Anlegen des Spiels und nach jedem Zug [broadcast] bzw. nach jedem Zug des Gegners [spielerspezifisch] dieses Set an Nachrichten rausgeschickt. Wenn das Spiel vorbei ist, einmalig ein „Game Over“</p>	/

command	command_key	Parameter	Rückgabewert	Beschreibung	Beispiel-Request
„play“	„create“	“command”: <ul style="list-style-type: none"> • “play” “command_key”: <ul style="list-style-type: none"> • “create” “game”: <ul style="list-style-type: none"> • “connect4” • “tictactoe” • “othello” • “nim” • “checkers” • “go” • “waldmeister” “mode”: <ul style="list-style-type: none"> • “player_vs_player” • “player_vs_ai” • “playerai_vs_ai” • “playerai_vs_playerai” “difficulty”: <ul style="list-style-type: none"> • “easy” • “medium” • “hard” 	<div> Success: <pre>"response_code": 200, "response_msg": "Game initialized"</pre> </div> <div> Error: <pre>"response_code": 253, "response_msg": "Arguments are missing or invalid!", "mode": "p", "difficulty": "hard" "response_code": 254, "response_msg": "Game still running. Please surrender first!" "response_code": 52, "response_msg": "Internal error occurred!"</pre> </div>	Initialisiert das Spiel im Backend mit der übergebenen Spielkonfiguration	<pre>{"command": "play", "command_key": "create", "game": "connect4", "mode": "player_vs_ai", "difficulty": "hard"}</pre>

command	command_key	Parameter	Rückgabewert	Beschreibung	Beispiel-Request
„play“	„valid_moves“	<p>“command”:</p> <ul style="list-style-type: none"> “play” <p>“command_key”:</p> <ul style="list-style-type: none"> “valid_moves” <p><i>Bei Spielen mit Spielsteinen verschieben zusätzlich die from_pos:</i></p> <p>“pos”:</p> <ul style="list-style-type: none"> “<int>” 	<p>Success:</p> <p>"response_code": 208, "response_msg": "Valid moves:", "moves": "[1, 2, 5]"</p> <p>Image <image></p> <p>Error:</p> <p>"response_code": 252, "response_msg": "You need to create a game first!" <i>[initial und wenn gegenwärtiges Spiel beendet]</i></p> <p>"response_code": 257, "response_msg": "Pos '<pos>' is not a pos!", "pos": "<pos>"</p> <p>"response_code": 257, "response_msg": "Pos must be greater than or equal to 0!"</p> <p>"response_code": 257, "response_msg": "Invalid from_pos!"</p>	<p>Zeigt valide Züge.</p> <p>Rückgabe als Bild mit gehighlighteten Feldern und als String mit validen Zugindizes.</p>	<pre>{ "command": "play", "command_key": "valid_moves", "pos": "<from_pos>" }</pre>

command	command_key	Parameter	Rückgabewert	Beschreibung	Beispiel-Request
„play“	„make_move“	“command”: <ul style="list-style-type: none"> “play” “command_key”: <ul style="list-style-type: none"> “make_move” “move”: <ul style="list-style-type: none"> “<move>” <ul style="list-style-type: none"> <i>Spiele, bei denen nur gesetzt wird:</i> to_pos: „<int >“ <i>Spiele, bei denen gezogen wird</i> from_pos und to_pos als Tupel: „(<int>, <int>)“ 	<div> Success: <pre> "response_code": 207, "response_msg": "Valid move." </pre> </div> <div> Error: <pre> "response_code": 252, "response_msg": "You need to create a game first!" [initial und wenn gegenwärtiges Spiel beendet] "response_code": 255, "response_msg": "'move' entry not set!" "response_code": 256, "response_msg": "Invalid move!" </pre> </div>	Führt Zug aus	<pre> {"command": "play", "command_key": "make_move", "move": "(<from_pos>, <to_pos>)"} </pre>

command	command_key	Parameter	Rückgabewert	Beschreibung	Beispiel-Request
„play“	„undo_move“	“command”: <ul style="list-style-type: none"> “play” “command_key”: <ul style="list-style-type: none"> “undo_move” “num”: <ul style="list-style-type: none"> “<int>” 	<div> Success: <pre> "response_code": 209, "response_msg": "Move successfully undone." </pre> </div> <div> Error: <pre> "response_code": 252, "response_msg": "You need to create a game first!" "response_code": 258, "response_msg": "Amount of moves to be undone not declared!" "response_code": 259, "response_msg": "Num: '<num>' is not an int!" "response_code": 259, "response_msg": " Amount of moves to be undone must be greater than 0!" </pre> </div>	<num> Züge rückgängig machen.	<pre> {"command": "play", "command_key": "undo_move", "num": "<num >"} </pre>

command	command_key	Parameter	Rückgabewert	Beschreibung	Beispiel-Request
„play“	„surrender“	“command”: <ul style="list-style-type: none"> “play” “command_key”: <ul style="list-style-type: none"> “surrender” 	Success: "response_code": 210, "response_msg": "Game over:" "result": "-1"	Stoppt laufendes Spiel und löscht dessen Spielinformation im Backend, darauffolgend dann entweder command_key „quit“ oder „new_game“	{"command": "play", "command_key": "surrender"}
			Error: "response_code": 252, "response_msg": "You need to create a game first!" <i>[initial und wenn gegenwärtiges Spiel beendet]</i>		
„play“	„quit“	“command”: <ul style="list-style-type: none"> “play” “command_key”: <ul style="list-style-type: none"> “quit” 	Success: "response_code": 211, "response_msg": "Game quit."	Führt Docker-Container herunter, trennt Verbindung von Server zu GameClient	{"command": "play", "command_key": "quit"}
			Error: "response_code": 254, "response_msg": "Game still running. Please surrender first!"		

command	command_key	Parameter	Rückgabewert	Beschreibung	Beispiel-Request
„play“	„new_game“	“command”: <ul style="list-style-type: none"> • “play” “command_key”: <ul style="list-style-type: none"> • “new_game” 	<div> Success: "response_code": 200, "response_msg": "Game initialized" </div> <div> Error: "response_code": 252, "response_msg": "You need to create a game first!" </div> <div> "response_code": 254, "response_msg": "Game still running. Please surrender first!" </div> <div> "response_code": 52, "response_msg": "Internal error occurred!" </div>	Initialisiert neues Spiel im Backend (in noch aktivem Docker- Container, mit aktiver GameClient- Verbindung)	{ "command": "play", "command_key": "new_game " }

command	command_key	Parameter	Rückgabewert	Beschreibung	Beispiel-Request
„play“	„blunder“	“command”: <ul style="list-style-type: none"> “play” “command_key”: <ul style="list-style-type: none"> “blunder” 	<div> Success: <pre> "response_code": 212, "response_msg": "No obvious blunder." "response_code": 213, "response_msg": "Blunder list (index, move):", "blunder": "[(0, 0), (8, 0)]" </pre> </div> <div> Error: <pre> "response_code": 252, "response_msg": "You need to create a game first!" </pre> </div>	Zeigt Züge, die nicht gut waren. Funktion für nach Spielende	{ "command": "play", "command_key": "blunder" }

command	command_key	Parameter	Rückgabewert	Beschreibung	Beispiel-Request
„play“	„timeline“	“command”: <ul style="list-style-type: none"> “play” “command_key”: <ul style="list-style-type: none"> “timeline” “num”: <ul style="list-style-type: none"> “<int>” 	<div> Success: <pre> "response_code": 214, "representation": <konsolenausgabe_board> Image <image> </pre> </div> <div> Error: <pre> "response_code": 252, "response_msg": "You need to create a game first!" "response_code": 260, "response_msg": "Timeline start index not declared!" "response_code": 261, "response_msg": "Index: '<num>' is not an int!" "response_code": 261, "response_msg": "Index must be greater than or equal to 0!" "response_code": 261, "response_msg": "Invalid timeline index!" </pre> </div>	Spiel nach Spielende noch einmal anschauen unter Angabe einer Zugnummer <num>. Wert 0 entspricht dem ersten Zug.	<pre> {"command": "play", "command_key": "timeline ", "num": "<num>"} </pre>

command	command_key	Parameter	Rückgabewert	Beschreibung	Beispiel-Request
„play“	„step“	“command”: <ul style="list-style-type: none"> “play” “command_key”: <ul style="list-style-type: none"> “step” 	<div> Success: <pre> "response_code": 214, "representation": <konsolenausgabe_board> </pre> Image <image> </div> <div> Error: <pre> "response_code": 252, "response_msg": "You need to create a game first!" "response_code": 261, "response_msg": "Invalid timeline index!" </pre> </div>	Innerhalb der Timeline um einen Zug nach vorne springen	{ "command": "play", "command_key": "step" }
„play“	„unstep“	“command”: <ul style="list-style-type: none"> “play” “command_key”: <ul style="list-style-type: none"> “unstep” 	<div> Success: <pre> "response_code": 214, "representation": <konsolenausgabe_board> </pre> Image <image> </div> <div> Error: <pre> "response_code": 252, "response_msg": "You need to create a game first!" "response_code": 261, "response_msg": "Invalid timeline index!" </pre> </div>	Innerhalb der Timeline um einen Zug nach hinten springen	{ "command": "play", "command_key": "unstep" }

command	command_key	Parameter	Rückgabewert	Beschreibung	Beispiel-Request
„play“	„evaluate“	“command”: <ul style="list-style-type: none"> “play” “command_key”: <ul style="list-style-type: none"> “evaluate” “game”: <ul style="list-style-type: none"> “connect4” “tictactoe” “othello” “nim” “checkers” “go” “waldmeister” “difficulty”: <ul style="list-style-type: none"> “easy” “medium” “hard” “num” <ul style="list-style-type: none"> “<int>” 	<div> Success: <pre> "response_code": 201, "response_msg": "Evaluation runs" "response_code": 203, "response_msg": "Evaluation finished:", "wins": 3, "losses": 1, "draws": 0 </pre> </div> <div> Error: <pre> "response_code": 254, "response_msg": "Game still running. Please surrender first!" "response_code": 262, "response_msg": "Num of games at evaluation not declared!" "response_code": 263, "response_msg": "Num: '<num>' is not an int!" "response_code": 263, "response_msg": "1 or more than 100 games not supported at evaluation!" </pre> </div>	<num> Spiele (Maximum: 100) spielen lassen ohne Anzeige. Statistische Auswertung über Anzahl Siege, Anzahl Niederlagen, und Anzahl Unentschieden. Möglichkeit, beim ersten der n Spiele sich Blunder und Timeline anzeigen zu lassen	<pre> {"command": "play", "command_key": "evaluate", "game": "connect4", "difficulty": "hard", "num": "<num>"} </pre>

			<pre>"response_code": 253, "response_msg": "Arguments are missing or invalid!", "difficulty": "hard" "response_code": 52, "response_msg": "Internal error occurred!"</pre>		
--	--	--	---	--	--

command	command_key	Parameter	Rückgabewert	Beschreibung	Beispiel-Request
„play“	„stop_evaluate“	“command”: <ul style="list-style-type: none"> “play” “command_key”: <ul style="list-style-type: none"> “stop_evaluate” 	<div> Success: <pre> "response_code": 203, "response_msg": "Evaluation finished:", "wins": 3, "losses": 1, "draws": 0 </pre> </div> <div> Error: <pre> "response_code": 252, "response_msg": "You need to create a game first!" </pre> </div>	Unterbrechung der Evaluation mit statistischer Auswertung der bisherig vollständig gespielten Spiele	{ "command": "play", "command_key": "stop_evaluate" }

command	command_key	Parameter	Rückgabewert	Beschreibung	Beispiel-Request
„exit“	/	“command”: <ul style="list-style-type: none"> “exit” 	/	Trennt die Verbindung vom Client zum Server	{ "command": "exit" }

command	command_key	Parameter	Rückgabewert	Beschreibung	Beispiel-Request
„debug“	„active_container“	“command”: <ul style="list-style-type: none"> “debug” “command_key”: <ul style="list-style-type: none"> “active_container“ 	<pre> "0": { ... } "1": { ... } "response_code": 300, "response_msg": "List of active GameClients:", "count": 2, </pre>	Entspricht docker container ls (Liste aller aktiven Container)	<pre> {"command": "debug", "command_key": "active_container" } </pre>
„debug“	„toggle_game_client_debug“	“command”: <ul style="list-style-type: none"> “debug” “command_key”: <ul style="list-style-type: none"> “toggle_game_client_debug“ 	<pre> "response_code": 301, "response_msg": "GameClient debug toggled.", "debug": true </pre>	Hält Docker-GameClient-Container offen, auch bei Fehlereintritt, wenn der GameClient intern geschlossen wird	<pre> {"command": "debug", "command_key": "toggle_game_client_debug"} </pre>