**Netzwerkprotokoll:**

CS: Client -> Server (Client -> ServerThreadForClient)

SC: Server-> Client (ServerThreadForClient -> ClientReaderThread)

Implemented

|  |  |  |  |
| --- | --- | --- | --- |
|  | Transmitted Keyword | Player input (only CS) lowercase/uppercase ignored | Description |
| SC | NAM1 |  | Server changes name |
| SC | QUIT |  | Player leaves the game |
| SC | HELP |  | Server shows list of all keywords on the clientside |
| CS | PLL1 | playerlist | Player wants to see who is connected to the server |
| SC | PLL2 String |  | Server shows a list of currently connected players |
| CS | GML1 | gamelist | Player wants to see open, ongoing and finished games |
| SC | GML2 String |  | gamelist will be printed |
| CS | HSC1 | highscore | Player wants to see high score |
| SC | HSC2 String |  | Server shows high score |
| CS | CHAT | chat | Player joins chat |
| CS | WHP1 String | /*playername* String | Player sends a whisperchat |
| SC | WHP2 String |  | received whisperchat is printed |
| SC | EWHP |  | Prints an ERROR-Message if the chosen playername does not exist. |
| CS | IDK | idk | Cheatcode: does nothing |
| CS | BRC1 | broadcastchat | Player enters Broadcast |
| SC | BRC2 |  | Server announces that the player is in broadcastchat |
| CS | QBR1 | quit broadcastchat | Player leaves Broadcast |
| SC | QBR2 |  | Server announces to the player is not in the broadcastchat anymore |
| CS | CRE1 | creategame | Player creates a new game with an individual game\_ID |
| SC | CRE2 String |  | Server creates game and shows that max. 4 players can be in this game and how many Players already are in this game |
| CS | JON1 String | join *game\_ID* | Player joins a created game from the lobby with the fitting game\_ID |
| SC | EJON String |  | Serves gives an ERROR if the game is full or/and started already or the game\_ID does not exist |
| SC | JON2 |  | Server announces that the player successfully joined the game and how many players are in the game |
| CS | STR1 | start | Starts the game |
| SC | STR2 |  | Informs Player that the game has started |
| SC | RNDS |  | Server shows how many rounds are left |
| SC | YTRN |  | Server announces if it is the players turn |
| SC | DICE |  | Server “rolls dice” and shows the steps the player got |
| CS | HXXD | up | Player moves a block up ingame |
| CS | DXXN | down | Player moves a block down ingame |
| CS | LXXT | left | Player moves a block left ingame |
| CS | RXXT | right | Player moves a block right ingame |
| SC | ERMO String |  | Server gives an ERROR if the player makes an invalid move |
| SC | POIN |  | Server shows the points the player gets for the landing on the block |
| SC | POIC |  | Server shows the points the player got for collecting the coin |
| SC | SCOR |  | Server shows the players score |
| SC | STPL |  | Server shows how many steps the player got left |
| SC | STPX |  | Server announces that the player got no more moves left |
| SC | EVEN |  | Server announces that an event is happening (flood or earthquake) |
| SC | DEAD |  | Server signals that the player is hit by an event and the player will return to the starting position |
| SC | MIPO |  | Server subtracts points from score when the player is hit by an event |
| SC | NTRN |  | Server announces the next turn |
| SC | RNDX |  | Server announces that there are no rounds left |
| SC | SCRE |  | Server announces that there is an even score between two or more players |
| SC | RNDA |  | Server adds an extra round |
| SC | WINX |  | Server announces winner |
| SC | HGHN |  | Server adds new high score |
| SC | ENDX |  | Server closes game |