

# Netzwerkprotokoll für Settlers of Asgard (Programmierprojekt CS108 Gruppe3) v3.0.0

## 1. Einleitung

Dieses Dokument beschreibt das Netzwerkprotokoll für das rundenbasierte Strategiespiel Settlers of Asgard. Es basiert auf TCP und verwendet ein textbasiertes, befehlsorientiertes Format mit fester Befehlslänge. Alle Nachrichten bestehen aus Befehlen und Parametern. Zwischen jedem Befehl und Parameter muss ein Dollar-Zeichen (\$) stehen.

## 2. Nachrichtenformat

Jede Nachricht hat folgendes Format:

`<Befehl>$<Parameter1>$<Parameter2>$...`

Beispiel:

`BILD$PL01$Castle$0001$0002` Dieser Befehl bedeutet, dass Spieler PL01 eine Struktur vom Typ "Castle" auf das Feld (1,2) bauen möchte.

**Feste Befehlslänge:** Jeder Befehl hat genau 4 Zeichen. Eine Ausnahme stellen die Nachrichten in der Fehlerbehandlung dar (`ERROR`, `OK`). Die Parameter haben eine variable Länge und werden durch Dollar-Symbole getrennt. Falls keine Parameter benötigt werden, bleibt die Nachricht nach dem Dollar-Symbol leer.

**Maximale Parameterlängen:**

- **Spielername:** maximal 16 Zeichen.
- **Nachricht:** maximal 200 Zeichen.
- Bei Überschreitung dieser Längen wird ein Fehler (Fehlercode 103) zurückgegeben.

## 3. Befehle

Spielverwaltung

Befehl	Richtung	Beschreibung	Parameter	Beispiel
RGST	C→S	Spieler verbindet sich erstmals mit Server	Spielername	RGST\$Thor0dinson
CHAN	C→S & S→C	Spieler verändert seinen Namen (S→C: Notify NameChange)	neuerSpielername	CHAN\$Thor0dinson
CREA	C→S	Neue Lobby erstellen	Spielername\$Lobbyname	CREA\$Thor0dinson\$Lobby1
JOIN	C→S	Spieler tritt Lobby bei	Spielername\$Lobbyname	CREA\$Lobby1\$Thor0dinson
LEAV	C→S	Spieler verlässt Lobby	Spielername\$Lobbyname	LEAV\$Thor0dinson\$Lobby1
EXIT	C→S	Spieler disconnected	Spielername\$	EXIT\$Thor0dinson\$

Befehl	Richtung	Beschreibung	Parameter	Beispiel
CHAN	C→S & S→C	Spieler möchte seinen Namen ändern	Spielername	CHAN\$LokiLaufeyson
CHTG	C→S	Nachricht senden (global)	Spielername\$Nachricht	CHTG\$ThorOdinson\$HelloWorld!
CHTL	C→S	Nachricht senden (lobby)	Spielername\$Nachricht	CHTL\$ThorOdinson\$HelloWorld!
CHTP	C→S	Nachricht senden (privat)	Spielername\$Spielername2\$Nachricht	CHTP\$PL01\$PL02\$HelloWorld!
LIST	C→S	Liste der Lobbies		LIST\$
LSTP	C→S	Liste der Spieler in der Lobby oder Serverweit	Ort (LOBBY oder SERVER)\$[lobbyName]	LSTP\$LOBBY\$lobbyName / LSTP\$SERVER\$
STRT	C→S	Spiel starten		STRT\$
STDN	S→C	Nachricht an Clients: Server wird beendet		STDN\$
PING	C→S & S→C	Verbindung prüfen		PING\$
GSTS	C→S	Spielstatus abrufen (temporär)		GSTS\$
GPRC	C→S	Preise für verschiedene Kaufbare Elemente abfragen (temporär)		GPRC\$
SYNC	C→S	Synchronisation des Spielzustands anfordern		SYNC\$
OK	S→C	Protokoll-Antwort bei erfolgreichen Befehlen	[Arg1] [\$Arg2]...	OK\$

## Spielmechanik

Befehl	Richtung	Beschreibung	Parameter	Beispiel
TURN	S→C	Startet Zug	Spielername	TURN\$ThorOdinson
ENDT	C→S	Beendet Zug		ENDT\$

Befehl	Richtung	Beschreibung	Parameter	Beispiel
ENDT	S→C	Beendet Zug eines Spielers (bei Zeitüberschreitung) -> zukünftig	Spielername	ENDT\$PL01
BUYT	C→S	Kauft ein Feld an Koordinaten x,y	X\$Y	BUYT\$1\$2
BYSR	C→S	Kauft eine Struktur	StrukturID	BYSR\$RuneTable
PLST	C→S	Baut Struktur an Koordinaten x,y	X\$Y\$StrukturID	PLST\$1\$2\$RuneTable
USSR	C→S	Eine Struktur verwenden	X\$Y\$StrukturID\$UseType	USSR\$1\$2\$RuneTable\$Runes
BYST	C→S	Kaufen einer Statue	StatueID	BYST\$Freyr
UPST	C→S	Aufwerten einer Statue	X\$Y\$StatueID	UPST\$1\$2\$Freyr
USTA	C→S	Benutzen einer Statue	X\$Y\$StatueID\$UseType	USTA\$1\$2\$Freyr\$Runes"
USPA	C→S	Benutzen eines Spieler-Artefakts (auf Spieler anwenden)	ArtifactID\$UseType	USPA\$1\$Runes
USFA	C→S	Benutzen eines Feld-Artefakts (auf Feld anwenden)	ArtifactID\$UseType	USFA\$2\$Runes
ENDG	S→C	Beenden des Spiels		ENDG\$

#### 4. Fehlerbehandlung

Eine allgemeine Fehlernachricht hat folgendes Format:

ERROR\$<Fehlercode>\$<Fehlermeldung>

Beispiel:

ERROR\$100\$Unbekannter Befehl

Spezifische Liste der Fehlercodes

Fehlercode	Name	Beschreibung	Beispiel
100	Unbekannter Befehl	Tritt auf, wenn der empfangene Befehl nicht im Protokoll definiert ist.	LEAV\$ThorOdinson\$SP01
101	Falsche Parameteranzahl	Tritt auf, wenn ein Befehl zu wenige oder zu viele Parameter enthält.	JOIN\$ThorOdinson\$SP01\$PL01
102	Ungültiger Parameter	Tritt auf, wenn ein Parameter nicht dem erwarteten Format entspricht.	JOIN\$12345\$SP01
103	Ungültige Länge	Tritt auf, wenn ein Befehl oder Parameter die maximal erlaubte Länge überschreitet.	JOIN\$ThorOdinson123456783456\$SP01;
104	Unzulässige Zeichen	Tritt auf, wenn unzulässige Zeichen (z. B. Semikolons) innerhalb eines Parameters gefunden werden.	JOIN\$Thor\$Odinson\$SP01
105	Unbekannte SpielerID	Tritt auf, wenn eine SpielerID (oder Spielername) nicht existiert oder ungültig ist.	TURN\$PL999
106	Nichtautorisierte Aktion	Tritt auf, wenn ein Spieler versucht, eine Aktion auszuführen, zu der er nicht berechtigt ist.	STRT\$PL02

Fehlercode	Name	Beschreibung	Beispiel
107	Spiel-ID nicht gefunden	Tritt auf, wenn eine angegebene Spiel-ID nicht existiert oder ungültig ist.	JOIN\$ThorOdinson\$SP999

### Protokoll-Antwort bei erfolgreichen Befehlen

Wenn ein Befehl korrekt verarbeitet wurde, schickt der Server eine Bestätigung zurück, damit der Client weiß, dass alles erfolgreich war. Diese Nachricht hat folgendes Format:

OK\$<Befehl>\$[Parameter1]\$[Parameter2]\$...

Die Parameter sind optional und hängen vom Befehl ab. Beispiel:

OK\$JOIN