



ulm university universität  
**uulm**

**Fakultät für  
Ingenieurwissenschaften,  
Informatik und  
Psychologie**

Institut für Software-  
technik und Program-  
miersprachen

# Softwaregrundprojekt Meilenstein 3

Softwaregrundprojekt an der Universität Ulm

**Vorgelegt von:**

Gruppe 10

**Dozent:**

Florian Ege

**Betreuer:**

Stefanos Mytilineos

2019

---

# Inhaltsverzeichnis

<b>1</b>	<b>Schnittstellenarten, Dialoge und Dialogstruktur</b>	<b>4</b>
1.1	Client . . . . .	4
1.1.1	Schnittstellenarten . . . . .	4
1.1.2	Dialoge . . . . .	5
1.1.3	Dialogstrukturdiagramme . . . . .	7
1.2	Server . . . . .	8
1.2.1	Schnittstellenarten . . . . .	8
1.2.2	Dialoge . . . . .	8
1.2.3	Dialogstruktur . . . . .	8
1.2.4	Zulässige Optionen . . . . .	8
1.3	Team- und Partiekonfigurator . . . . .	9
1.3.1	Schnittstellenarten . . . . .	9
1.3.2	Dialoge . . . . .	9
1.3.3	Dialogstrukturdiagramme . . . . .	10
1.4	KI-Client . . . . .	10
1.4.1	Schnittstellenarten . . . . .	10
1.4.2	Dialoge . . . . .	10
1.4.3	Dialogstruktur . . . . .	11
1.4.4	Zulässige Optionen: . . . . .	12
<b>2</b>	<b>Grafische Gestaltung und Nutzungskonzept</b>	<b>13</b>
2.1	Client . . . . .	13
2.1.1	Spielansicht für einen Spieler . . . . .	13
2.1.2	Spielansicht für einen Beobachter . . . . .	14
2.1.3	Hauptmenü . . . . .	15
2.1.4	Spielsuche . . . . .	15
2.1.5	Hilfe . . . . .	16
2.1.6	Hotkeys . . . . .	17
2.1.7	Team ändern . . . . .	17
2.1.8	Bestätigungsaufforderung . . . . .	18
2.1.9	Fehler . . . . .	18
2.1.10	Pause . . . . .	19
2.1.11	Verbindungsabbruch . . . . .	19
2.1.12	Spielende . . . . .	20
2.2	Server . . . . .	20
2.2.1	ServerInit . . . . .	20
2.2.2	ServerRunning . . . . .	20
2.2.3	ServerError . . . . .	21
2.3	Team- und Partiekonfiguraton . . . . .	21
2.3.1	Konfiguratormenü . . . . .	21

---

2.3.2	Teammenü . . . . .	22
2.3.3	Team bzw. Partiekonfiguration laden . . . . .	22
2.3.4	Teamkonfigurator . . . . .	23
2.3.5	Team bzw. Partiekonfiguration speichern . . . . .	24
2.3.6	Konfiguration erfolgreich . . . . .	24
2.3.7	Konfiguration ungültig . . . . .	25
2.3.8	Partiemenü . . . . .	25
2.3.9	Partiekonfigurator . . . . .	26

---

# 1 Schnittstellenarten, Dialoge und Dialogstruktur

## 1.1 Client

Der Client stellt die Anwendung dar, mit der ein Nutzer aktiv als Spieler oder auch passiv als Beobachter an einer Partie teilnehmen kann.

### 1.1.1 Schnittstellenarten

Als Benutzerschnittstelle wird eine grafische Benutzeroberfläche verwendet.

**Begründung:** Um das Spiel so intuitiv wie möglich zu gestalten, ist es sinnvoll, dem Nutzer alle für das Spielgeschehen relevanten Informationen und Aktionen grafisch in einer GUI darzustellen. Hinzu kommt, dass neben dem eigentlichen Spiel auch die zugehörigen Funktionen, wie zum Beispiel das Verbinden mit einem Server, benutzerfreundlich und leicht zu bedienen sein sollte. Dies lässt sich am leichtesten durch eine grafische Benutzeroberfläche bewerkstelligen.

---

### 1.1.2 Dialoge

Im Folgenden sind alle Dialoge, die während der Nutzung des Clients benötigt werden, den zugehörigen Anwendungsfällen zugeordnet.

Name	Typ	Abgedeckte Anforderungen	
Startbildschirm	Dialog	FA60	Hauptmenü [Ansicht]
Hilfe	Dialog	FA65	Hilfe [Ansicht]
		QA18	Benutzerfreundlichkeit
Hotkey-Liste	Dialog	FA65	Hilfe [Ansicht]
		FA68	Hotkeys
Team ändern	Popup	FA63	Team-Konfiguration importieren [Ansicht]
		FA15	Teams
		FA54	Quidditchtem-Konfiguration
		FA61	Spiel beitreten [Ansicht]
Beenden	Popup		
Spielstart fehlgeschla- gen	Popup	QA16	Zuverlässigkeit
Verlassen	Popup		
Spielsuche	Dialog	FA61	Spiel beitreten [Ansicht]
		FA55	Netzwerkschnittstelle
Spielende	Dialog	FA62	Spiel Ende [Ansicht]
		FA49	Spielende
Beobachten	Dialog	FA66	Beobachter [Ansicht]
Verbindungsabbruch	Dialog	QA16	Zuverlässigkeit
		QA17	Robustheit
Pause	Dialog	FA69	Pausieren

---

Spiel	Dialog	FA64	Spiel[Ansicht]
		FA1	Spielfeld
		FA2	Mittelkreis
		FA3	Mittelzelle
		FA4	Hüterzone
		FA5	Zelle
		FA6	Torring
		FA8	Punkte erzielen
		FA10	Ball
		FA11	Quaffel [Ball]
		FA12	Klatscher [Ball]
		FA13	Goldener Schnatz [Ball]
		FA14	Besen
		FA16	Spielfiguren
		FA17	Jäger
		FA18	Treiber
		FA19	Hüter
		FA20	Sucher
		FA31	Fans
		FA32	Elfen [Fantyp]
		FA33	Kobolde [Fantyp]
		FA34	Trolle [Fantyp]
		FA35	Niffler [Fantyp]
		FA36	Schiedsrichter
		FA44	Runde
		FA46	Spielerphase

### 1.1.3 Dialogstrukturdiagramme



---

## 1.2 Server

Diese Komponente hostet Spiele und enthält die Spiellogik. Zu Beginn wird der Server einmalig vom Systemadministrator gestartet und steht anschließend den Spielern von *Fantastic Feasts* zur Verfügung.

### 1.2.1 Schnittstellenarten

Als Benutzerschnittstelle wird ein CLI verwendet. **Begründung:** Der Nutzer kommt mit dieser Komponente über keine Benutzerschnittstelle in Berührung. Deswegen spielen Look and Feel keine Rolle. Zusätzlich wird der Server nur einmal mit wenigen Parametern gestartet, benötigt zur Laufzeit dann keine weiteren Eingaben und muss auch keinerlei graphische Ausgabe zur Verfügung stellen, weswegen das CLI ausreicht.

### 1.2.2 Dialoge

Im Folgenden werden die Anforderungen an den Server einem CLI-Dialog zugeordnet.

Name	Typ	Abgedeckte Anwendungsfälle	
ServerInit	CLI Befehl mit Params	FA57	Partie-Konfiguration laden
ServerRunning	Response	FA58, FA59	Spielmechanik.
ServerError	Response	allgemein	Feedback

### 1.2.3 Dialogstruktur

Die Dialogstruktur des Servers lässt sich wie folgt beschreiben: **ServerInit:** Der Server lässt sich in der Konsole mit dem Namen der Server-Anwendung, dem Namen einer gültigen Partie-Konfiguration und einem Parameter aufrufen. Der Parameter ist die Port-Nummer, über die der Server erreichbar ist. Darauf gibt es zwei möglichen Antworten. **ServerRunning:** War die Initialisierung erfolgreich, antwortet der Server mit einer entsprechenden Nachricht. **ServerError:** Im Falle eines Fehlers bei der Initialisierung wird mit einer Fehlernachricht geantwortet.

### 1.2.4 Zulässige Optionen

Flag	Erklärung
-p	Legt die Portnummer fest.
--help	Zeigt eine Liste möglicher Optionen an.



---

## 1.3 Team- und Partiekonfigurator

Diese Komponente enthält einen Konfigurator, mit dem sich sowohl Quidditch-Teams, als auch Partie-Konfigurationen erstellen und bearbeiten lassen.

### 1.3.1 Schnittstellenarten

Als Benutzerschnittstelle wird, wie im Lastenheft vorgeschrieben, eine GUI verwendet.

**Begründung:** Der Nutzer möchte alle Informationen zu einer gegebenen Konfiguration übersichtlich dargestellt bekommen und anhand dieser Darstellung direkt Änderungen vornehmen können. Hierfür ist eine GUI die intuitivste und sinnvollste Variante, da eine grafische Darstellung eine übersichtliche Visualisierung erlaubt und eine Änderung direkt anhand dieser Visualisierung möglich ist.

### 1.3.2 Dialoge

Im Folgenden werden die bereits formulierten Anforderungen und Anwendungsfälle der Komponente den entsprechenden Dialogen zugeordnet.

Name	Typ	Abgedeckte Anwendungsfälle	
Konfiguratorermenü	Dialog	QA18	implizit aus Benutzerfreundlichkeit
Teammenü	Dialog	QA18	implizit aus Benutzerfreundlichkeit
Team laden	Dialog	FA72	Konfiguration öffnen
Teamkonfigurator	Dialog	FA70-72	Konfiguration erstellen/bearbeiten
Team speichern	Dialog	FA72	Konfiguration speichern
Partiemenü	Dialog	–	implizit, da Strukturierung erforderlich
Partiekonfiguration laden	Dialog	FA72	Konfiguration öffnen
Partiekonfigurator	Dialog	FA70-72	Konfiguration erstellen/bearbeiten
Partiefonfiguration speichern	Dialog	FA72	Konfiguration speichern
Konfiguration erfolgreich	Popup	QA17-18	Benutzerfreundlichkeit und Robustheit
Konfiguration ungültig	Popup	QA17-18	Benutzerfreundlichkeit und Robustheit

### 1.3.3 Dialogstrukturdiagramme



## 1.4 KI-Client

Diese Komponente simuliert einen Menschlichen Gegner, der sich wie ein normaler Client bei einem Spielserver anmeldet und dann autonom seine Spielentscheidungen trifft.

### 1.4.1 Schnittstellenarten

Es besteht kein Grund, für diese Komponente eine grafische Oberfläche bereitzustellen, da die Anwendung zur Laufzeit keine Eingabe von einem menschlichen Benutzer erwartet und eine Partie mittels des Clients verfolgt werden kann.

Für eine Kommandozeilenanwendung ist es einfacher, Plattformunabhängigkeit sicherzustellen. Außerdem wird es damit problemlos möglich, den KI-Client aus einem anderen Programm zu starten. Beispielsweise kann dem Client eine Funktion hinzugefügt werden, gegen die KI zu spielen, ohne dass der Benutzer den KI-Client extern starten muss.

### 1.4.2 Dialoge

Im Folgenden werden die Anforderungen an den KI-Client einem CLI-Dialog zugeordnet.

Name	Typ	Abgedeckte Anwendungsfälle	
Init	CLI Befehl mit Params	FA73 - FA75	Schwierigkeit, Server- und Team-Konfiguration einstellen.
InitFailure	Response	FA73 - FA75	Feedback.
WaitingForGame	Response	FA55	Netzwerkschnittstelle, allgemeine Kommunikation.
Playing	Response	allgemein	Feedback.
AttemptingReconnect	Response	FA55	Netzwerkschnittstelle, allgemeine Kommunikation.
ConnectionLost	Response	FA55	Netzwerkschnittstelle, allgemeine Kommunikation.

### 1.4.3 Dialogstruktur

**Init:** Der KI-Client wird über die Kommandozeile gestartet. Serverkonfiguration, Team-Konfiguration, die maximale Anzahl an Reconnect-Versuchen und der Schwierigkeitsgrad werden beim Start der Anwendung mittels Kommandozeilenparametern gehandhabt. Der Server, mit dem sich der KI-Client verbinden soll wird als Argument übergeben. Die Team-Konfiguration und der Schwierigkeitsgrad können mittels Optionen verändert werden und nehmen ansonsten einen Standardwert an.

**InitFailure:** Wird eine ungültige Option angegeben, ist der Server nicht erreichbar oder wurde eine ungültige Team-Konfigurationsdatei geladen, so erscheint eine Fehlermeldung mit entsprechenden Hinweisen.

**WaitingForGame:** War die Initialisierung erfolgreich und der KI-Client konnte sich mit dem Server verbinden, so erscheint eine Nachricht, die darauf hinweist, dass noch auf den Beginn der Partie gewartet wird.

**Playing:** Während einer laufenden Partie wird ein Hinweis angezeigt.

**AttemptingReconnect:** Bei einem Verbindungsabbruch zeigt der KI-Client eine Meldung an und versucht automatisch, die Verbindung wiederherzustellen.

**ConnectionLost:** Konnte sich der KI-Client nicht innerhalb der angegebenen Anzahl von Versuchen erneut mit dem Server verbinden, so erscheint eine Fehlermeldung und das Programm beendet sich.

Der KI-Client beendet sich außerdem nach Abschluss einer Partie durch ein reguläres Spielende.

---

#### 1.4.4 Zulässige Optionen:

Flag	Erklärung
-s	Legt den Schwierigkeitsgrad fest. Akzeptiert eine ganze Zahl zwischen 0 und 2, wobei 0 für einfach, 1 für mittelschwer und 2 für schwer steht. Bei einer ungültigen Eingabe wird eine Fehlermeldung ausgegeben.
-t	Legt die Team-Konfiguration fest. Akzeptiert einen String als Pfad zu einer JSON-Datei. Existiert der angegebene Pfad nicht oder ist die Datei keine gültige Konfigurationsdatei, wird eine entsprechende Fehlermeldung ausgegeben.
-r	Reconnects: Spezifiziert, wie oft der KI-Client versucht, die Verbindung nach einem Verbindungsabbruch wiederherzustellen. Standardwert ist 5.
--help	Zeigt eine Liste möglicher Optionen an.

## 2 Grafische Gestaltung und Nutzungskonzept

### 2.1 Client

#### 2.1.1 Spielansicht für einen Spieler



In der Spielansicht kann ein Spieler das aktuelle Spielgeschehen verfolgen und seine Züge ausführen. Dabei ist die Oberfläche in mehrere Teile unterteilt.

Im *Stats* Bereich werden grundlegende Informationen über die aktuelle Partie und die beiden Teams dargestellt. Das Team, das an oberster Position steht, ist momentan am Zug. Im darunter liegende Bereich wird die *aktuell ausgewählte Spielfigur* hervorgehoben. Zusätzlich wird aufgeführt, um welchen Typen von Spielfigur es sich handelt und welche grundlegenden Züge diese Figur ausführen kann. In den Fan-Phasen werden hier vergleichbare Informationen zu den Fans angezeigt. Darunter befinden sich drei Buttons mit dem entweder das Spiel *pausiert* werden, alle Veränderungen, die man in der aktuellen Runde getätigt hat *zurücksetzen* oder seinen Zug endgültig *ausführen* kann. Am unteren Rand der Oberfläche ist eine *Legende* mit einer Übersicht über alle Spielfiguren des eigenen und des gegnerischen Teams zu sehen. Daneben befindet sich ein Feld, in dem Statusmeldungen angezeigt werden können. Beispiele für solche Statusmeldungen sind eine Benachrichtigung über ein Foul oder über das erfolgreiche Ausführen eines Spielzuges. Den größten Teil der Oberfläche nimmt das eigentliche Spielfeld ein. Hier

werden alle Spielfiguren in den Feldern angezeigt, auf denen sie sich gerade befinden. Ist man am Zug, so werden alle Züge, die von der aktuell ausgewählten Spielfigur ausgeführt werden können, farblich hervor gehoben. Der Spieler kann diese Aktionen dann durch Klicken auf die passende Zelle ausführen. Ist man nicht am Zug, so werden alle Eingabemöglichkeiten, mit Ausnahme des *Pause* Buttons, deaktiviert.

## 2.1.2 Spielansicht für einen Beobachter



In der Spielansicht kann ein Beobachter eine Partie zwischen zwei anderen Gegnern passiv verfolgen. Die Oberfläche ist im wesentlichen gleich aufgebaut wie die Oberfläche, die die Spieler sehen. Jedoch sind beim Beobachter alle Felder, die zur Eingabe dienen, deaktiviert. Die einzige Interaktion, die durch einen Button ermöglicht wird, ist das vorzeitige *Verlassen* einer Partie.

---

### 2.1.3 Hauptmenü



Dieser Dialog erscheint nach dem Start der Anwendung. Der „Spielen“-Button öffnet den Spielsuche-Dialog. Der „Team ändern“-Button öffnet das „Team ändern“-Popup. Der „Hilfe“-Button öffnet den Hilfe-Dialog. Der „Beenden“-Button öffnet ein Bestätigungs-Popup und beendet bei positiver Antwort die Anwendung.

### 2.1.4 Spielsuche



---

Der Benutzer gibt zuerst die Adresse und den Port des Spielervers ein, mit dem er sich verbinden möchte. Wenn er die Partie beobachten will, wählt er „Als Gast beitreten“ aus. Drückt er anschließend auf den „Start“-Button, versucht sich der Client mit dem angegebenen Server zu verbinden. Mit dem „Zurück“-Button gelang man zurück auf den Startbildschirm.

### 2.1.5 Hilfe

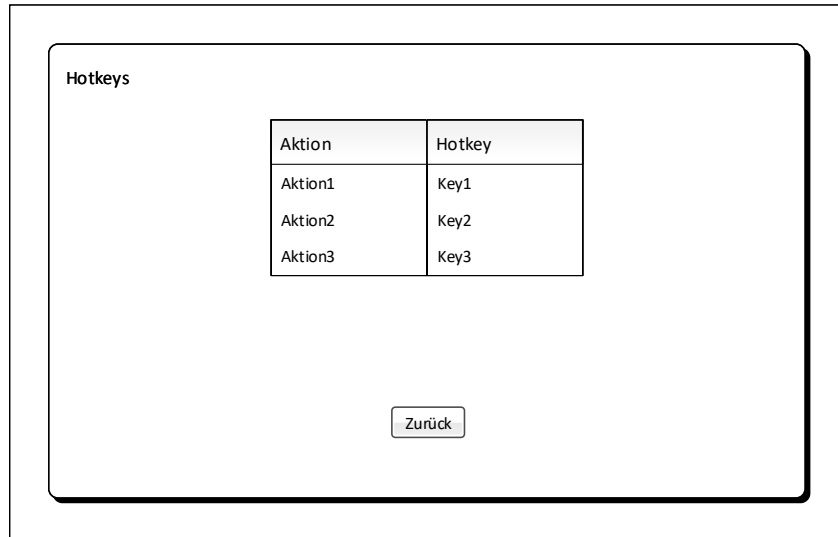


In einem Textfeld, gegebenenfalls mit Scrollbar, wird ein Hilfetext angezeigt. Der „Zurück“-Button öffnet den Startbildschirm, der „Hotkey“-Button den Hotkey-Dialog.



---

## 2.1.6 Hotkeys



Hier werden alle verfügbaren Hotkeys in Tabellenform aufgelistet. Der „Zurück“-Button öffnet den Hilfe-Dialog.

## 2.1.7 Team ändern



In diesem Popup kann der Benutzer durch sein Dateisystem navigieren und eine JSON-Datei auswählen. Hat er eine gültige Datei ausgewählt und betätigt den „Auswählen“-Button, wird das Popup geschlossen und die Team-Konfiguration für das Spiel verwendet. Der „Abbrechen“-Button schließt das Popup und es werden keine Änderungen vorgenommen.

---

### 2.1.8 Bestätigungsaufforderung



Dieser Aufbau wird für die Popups „Beenden“ und „Verlassen“ verwendet. Der „Abbrechen“-Button schließt das Popup und der Benutzer gelangt zurück in den Dialog, in dem er vor Öffnen des Popups war. Der „OK“-Button führt dazu, dass eine Aktion ausgeführt wird.

### 2.1.9 Fehler



Dieser Aufbau wird für das Popup „Spielstart fehlgeschlagen“ verwendet. Der angezeigte Text richtet sich nach dem aufgetretenen Fehler. Der „OK“-Button schließt das Popup.

---

### 2.1.10 Pause



Wird angezeigt, wenn das Spiel von einem Spieler pausiert wird. Der „Weiter“-Button setzt die Partie fort und kann nicht von einem Gast betätigt werden.

### 2.1.11 Verbindungsabbruch



Im Falle eines Verbindungsabbruchs zwischen Client und Server wird dieser Dialog angezeigt. Wird die Verbindung wiederhergestellt, gelangt der Benutzer automatisch wieder zurück in den vorherigen Dialog. Alternativ kann er durch betätigen des Buttons zum Startbildschirm gelangen. Ist er ein Spieler, kann er die Partie nicht weiterführen und sein Gegner gewinnt nach Ablauf einer Zeitdauer die Partie.

---

### 2.1.12 Spielende



Bei Spielende wird dieser Dialog geöffnet. Hier werden der Punktestand bei Spielende und der Name des Gewinners angezeigt. Der Button öffnet den Startbildschirm-Dialog.

## 2.2 Server

### 2.2.1 ServerInit

```
$ server -p1230 standard_partie.json
... pending
```

Das Beispiel stellt einen Aufruf des Servers mit entsprechenden Argumenten dar. Mit dem Parameter wird eine Port-Nummer angegeben. Das zweite Argument gibt den Pfad zu einer gültigen Partie-Konfigurationsdatei an.

### 2.2.2 ServerRunning

```
Fantastic Feasts server is running on Port 1230 ...
```

Im Falle einer erfolgreichen Initialisierung wird dem Systemadministrator diese Nachricht mitgeteilt. Es können hier auch weitere Informationen über den initialisierten Server ausgegeben werden.

---

### 2.2.3 ServerError

Error: Initialization failed

Im Falle eines Fehlers bei der Initialisierung wird der Systemadministrator mit dieser Fehlermeldung benachrichtigt.

## 2.3 Team- und Partiekonfigurator

### 2.3.1 Konfiguratormenü



Über das Konfiguratormenü kann eine Auswahl zwischen dem Teammenü und dem Partiemenu über die entsprechenden Buttons getroffen werden. Diese führen zu den Menüs der jeweiligen Konfiguratoren. Über den Button *Verlassen* kann der Konfigurator verlassen werden.

---

### 2.3.2 Teammenü



Über das Teammenü kann eine Auswahl zwischen dem Laden und dem Erstellen einer Teamkonfiguration getroffen werden. Das erfolgt über die entsprechenden Buttons. Über den Button *Verlassen* kann das Teammenü verlassen werden.

### 2.3.3 Team bzw. Partiekonfiguration laden



Über diesen Dialog kann eine bereits vorhanden Konfigurationsdatei ausgewählt und im Konfigurator geladen werden. Das erfolgt über ein Dateiauswahlelement und die entsprechenden Buttons. Über den Button *Abbrechen* gelangt man zurück zu den entsprechenden Menüs. Da das Laden einer Teamkonfiguration nahezu identisch zum Laden einer Partiekonfiguration ist, wurde diese beiden Fälle in einem zusammen gefasst. Sollte eine zu ladende Konfigurationsdatei ungültig sein, öffnet sich das Popup *Konfiguration ungültig*.

### 2.3.4 Teamkonfigurator

Teamkonfigurator

Teamname

Teamfarbe

Teamfarbe (Ersatz)

Teamlogo

Fan

Fan

Fan

Fan

Fan

Fan

Fan

Fan

Fan

Fan

Fan

Speichern

Abbrechen

Jäger

Treiber

Hüter

Sucher

Name

männlich

weiblich

Besen

Name

männlich

weiblich

Besen

Name

männlich

weiblich

Besen

Name

männlich

weiblich

Besen

Name

männlich

weiblich

Besen

Name

männlich

weiblich

Besen

Name

männlich

weiblich

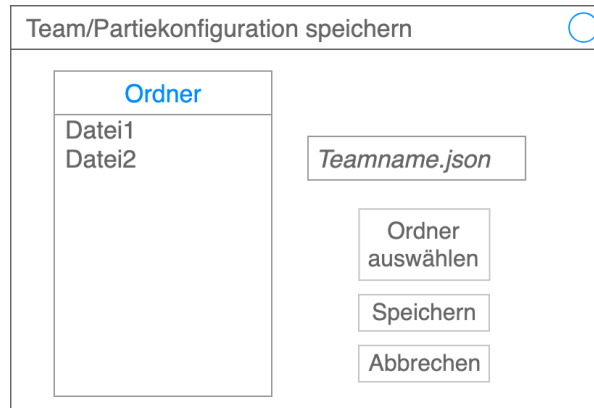
Besen

Im Teamkonfigurator können alle Parameter eines Teams eingestellt werden. Team- und Spielernamen lassen sich durch ein Textfeld bearbeiten. Teamfarben sind über eine Farbauswahl einstellbar. Das Teamlogo lässt sich aus einer Liste vorhandener Logos auswählen. Die Kästen mit den Kreuzen dienen als Platzhalter für Icons, die die verschiedenen Fans und Spieler darstellen und somit unterscheidbar machen. Fans sowie Besen der Spieler sind über eine Dropdown-Auswahl einstellbar. Das Geschlecht der Spieler lässt sich über Radio-Buttons einstellen. Bei jeder Änderung werden die entsprechenden Bedingungen für eine gültige Konfiguration geprüft und der Nutzer erhält visuelles Feedback (z. B. in Form von roter Schriftfarbe in den entsprechenden Feldern). Über den Button *Abbrechen* lässt sich der Konfigurator jederzeit verlassen. Ist eine gültige Auswahl eingestellt, kann die Konfiguration über *Speichern* in einem separaten Speicherdialog persistiert werden.

23

---

### 2.3.5 Team bzw. Partiekonfiguration speichern

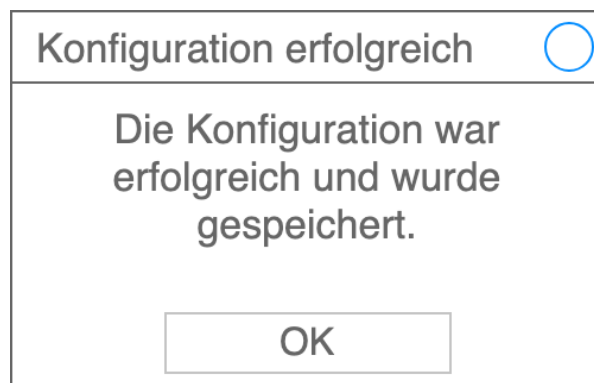


The dialog box is titled "Team/Partiekonfiguration speichern" and features a close button (blue circle) in the top right corner. It contains a list box on the left labeled "Ordner" with two entries, "Datei1" and "Datei2". To the right of the list box is a text input field containing "Teamname.json". Below the input field are three buttons: "Ordner auswählen", "Speichern", and "Abbrechen".

In diesem Dialog kann Dateiname und Speicherort der Konfiguration festgelegt werden. Der Button *Abbrechen* bringt den Nutzer direkt zurück in den entsprechen Konfigurator. Durch Klicken auf den Button *Speichern* wird die Datei mit dem gewählten Namen und Speicherort gespeichert.

Wurde versucht eine Konfiguration mit ungültigen Parametern zu speichern oder trat beim Speichervorgang ein Fehler auf, wird der Nutzer über dieses Popup darüber informiert. Der *Ok*-Button führt zurück zum entsprechenden Konfigurator.

### 2.3.6 Konfiguration erfolgreich



The dialog box is titled "Konfiguration erfolgreich" and features a close button (blue circle) in the top right corner. The main text area contains the message "Die Konfiguration war erfolgreich und wurde gespeichert." Below the message is a single button labeled "OK".

Wenn alle Paramter einer Konfiguration gültig waren, wird dieser Dialog angezeigt. Der *Ok*-Button führt zurück zum entsprechenden Menü.



---

### 2.3.7 Konfiguration ungültig



Wurde versucht eine Konfiguration mit ungültigen Parametern zu speichern oder trat beim Speichervorgang ein Fehler auf, wird der Nutzer über dieses Popup darüber informiert. Der *Ok*-Button des Popups führt zurück zum entsprechenden Dialog.

### 2.3.8 Partimenü



Über das Partimenü kann eine Auswahl zwischen dem Laden und dem Erstellen einer Partiekonfiguration getroffen werden. Das erfolgt über die entsprechenden Buttons. Über den Button *Verlassen* kann das *Partimenü* verlassen werden.

### 2.3.9 Partiekonfigurator

Partiekonfigurator

Partikonfigname

Rundenanzahl bis Überlänge

100

Zeitspanne für Rundenaktionen

Aktion1

100 s

Aktion2

100 s

Ereigniswahrscheinlichkeiten

Ereignis1

0,15

Ereignis1

0,50

Ereignis1

0,33

Weitere Parameter können folgen...

Speichern

Abbrechen

Im Partiekonfigurator können alle Parameter für eine gültige Partiekonfigurationsdatei eingestellt werden. Dazu gehören unter anderem die Rundenanzahl, bis Überlänge erreicht ist, die Zeitspannen für die jeweiligen Spielaktionen und Ereigniswahrscheinlichkeiten. Je nach Art des Parameters sind Spinner, Slider, Textfelder oder im weiteren Verlauf der Implementierung noch andere Auswahlelemente vorhanden.

Über den Button *Speichern* gelangt man in den *Partiekonfiguration speichern*-Dialog. Der Button *Abbrechen* führt zurück ins *Partiemenü*.