

# **Boardgame Simulator Anleitung**

February 17, 2021



# Contents

<b>1</b>	<b>Spiel erstellen</b>	<b>5</b>
1.1	Die Datei <i>game.xml</i> . . . . .	5
1.2	Die Datei <i>game_instance.xml</i> . . . . .	5
1.3	Spielobjekttypen . . . . .	7
1.3.1	Der Typ <i>card</i> . . . . .	8
1.3.2	Der Typ <i>figure</i> . . . . .	8
1.3.3	Der Typ <i>dice</i> . . . . .	9
<b>2</b>	<b>Spielsteuerung</b>	<b>11</b>



# 1 Spiel erstellen

Jedes Spiel besteht aus Spielobjekten (Bildern im jpg oder png Format) und den XML-Dateien *game.xml* und *game\_instance.xml*. Spielobjekte können drei verschiedene Typen haben *card*, *figure* und *dice*.

## 1.1 Die Datei *game.xml*

In der *game.xml* werden die Spielobjekte definiert. Das Grundgerüst der Datei sieht wie folgt aus.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<xml>
<background>hintergrund.jpg</background>
</xml>
```

Das Bild mit dem Namen *hintergrund.jpg* wird zum Hintergrundbild des Spiels. Zusätzlich können beliebige Spielobjekte durch den Tag

```
<object type="card"></object>
```

mit den unten genannten Typen definiert werden (in diesem Fall vom Typ *card*). Welche Attribute die verschiedenen Spielobjekte haben können wird in den entsprechenden Abschnitten erläutert.

## 1.2 Die Datei *game\_instance.xml*

In der *game\_instance.xml* wird eine Instanz des Spiels definiert, z.B. ob es einen Tisch geben soll oder nicht und an welcher Stelle die Objekte im Spiel zu Beginn liegen.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<xml>
<settings>
  <name>Brettspiel</name>
  <table color="#d2a56d" put_down_area="true"
    table_radius="500">true</table>
</settings>
<object unique_name="object1" x="0" y="0" r="0"/>
</xml>
```

Das Beispiel definiert ein Spiel mit dem Namen *Brettspiel*, das ein Tisch mit Radius 500, der Farbe *#d2a56d*, und einem Ablagebereich in der Mitte des Tisches hat. Das

## 1 Spiel erstellen

Spiel hat außerdem ein Objekt mit dem Namen *objekt1*, welches in *game.xml* definiert wurde und an Position 0,0 und mit Anfangsrotation 0 gezeichnet wird.

Unter dem Tag *settings* können alle wichtigen Spieleigenschaften definiert werden. Unter dem Tag *object* werden die Objekte mit Positionen, Rotationen usw. definiert.

Tagname	Attribute	Werte	Erklärung
name	–	String	Name des Spiels
table	color, put_down_area, table_radius	true/false	Anzeige des Tisches auf dem Spielfeld, falls False wird der Tisch nicht angezeigt, color definiert die Farbe des Tisches, put_down_area ist ein boolsches Attribut und gibt an, ob ein Ablagebereich in der Mitte des Tisches erscheinen soll, table_radius gibt den Radius des Tisches an.
private_area	–	true/false	gibt an, ob der Handkartenbereich zum Spielstart eingeblendet sein soll
seats	–	<seat color="#0000ff"/>	Feste Liste von farbigen Stühlen um den Tisch
debug_mode	–	true/false	Falls true, werden zusätzliche Informationen angezeigt, die hilfreich beim debuggen sind

Table 1.1: Mögliche Spiel Settings

## 1.3 Spielobjekttypen

Die Spielobjekte werden in der *game.xml* definiert. Einzelne Instanzen von Spielobjekten werden in der *game\_instance.xml* definiert und mit einem eindeutigen Namen mit den Objekten aus der *game.xml* verknüpft. Spielobjekte werden durch den Tag *object* markiert. Alle Spielobjekte haben die folgenden Attribute.

Attribute	Erklärung
type	Typ des Spielobjekts
unique_name	Eindeutiger Name des Objekts
width	Breite des Objekts
height	Höhe des Objekts

Weitere Attribute für *card*, *figure*, *dice* Objekte können in den jeweiligen Abschnitten gefunden werden.

Instanzen von Spielobjekten können in der *game\_instance.xml* mit folgenden Attributen definiert werden:

Attribute	Erklärung
unique_name	eindeutiger Name aus der <i>game.xml</i>
x	x Position zu Spielbeginn
y	y Position zu Spielbeginn
r	Rotation
s	Skalierung
number	Anzahl der Objekte des Typs
is_fixed	true/false, falls true kann Objekt im Spiel nicht bewegt werden

Beispieldefinition von Instanzen des Objekts mit dem Namen *objekt1* in der *game\_instance.xml*:

```
<object unique_name="objekt1" x="0" y="0" r="0" s="1"
number="3" is_fixed="false"/>
```

In diesem Beispiel werden 3 Instanzen vom Objekt mit dem Namen *objekt1* an der Position 0,0 mit der Rotation 0 und ohne Skalierung erzeugt.

### 1.3.1 Der Typ **card**

Objekte vom Typ **card** sind Spielkarten. Sie können z.B. gestapelt und gemischt werden und haben eine Vorder- und Rückseite. Beispiel eines **card** Objekts mit Vorderseite *front.jpg*, Rückseite *back.jpg*, dem Wert 10 hat und in 45 Grad Schritten gedreht werden kann und zur Kartengruppe *Spielkarte* gehört:

```
<object type="card" unique_name="card1" value="10"
  rotation_step="45" front="front.jpg" back="back.jpg">
  <group>Spielkarte</group>
</object>
```

Attribute	Optional	Erklärung
front	Nein	Dateiname der Vorderseite (jpg, png)
back	Nein	Dateiname der Rückseite (jpg, png)
value	Ja	Wert der Karte
rotation_step	Ja	Mögliche Rotationen der Karte

Objekte vom Typ **card** können als Wert eine Gruppe erhalten über die sie eindeutig identifizierbar und sammelbar sind.

```
<group>Spielkarte</group>
```

### 1.3.2 Der Typ **figure**

Objekte vom Typ **figure** sind Spielfiguren. Sie können z.B. stehen oder liegen und auf Karten gestellt werden.

Beispiel eines **figure** Objekts:

```
<object type="figure" unique_name="objekt1"
  standing="standing.jpg" width="100"
  height="100"/>
```

Attribute	Optional	Erklärung
standing	Nein	Bild für die stehende Spielfigur (jpg, png)



### 1.3.3 Der Typ **dice**

Objekte vom Typ **dice** sind Würfel. Sie haben mehrere Seiten und können gewürfelt werden.

Beispiel eines **dice** Objekts mit sechs Seiten:

```
<object type="dice" unique_name="dice1" width="30" height="30">
  <side value="1">side1.jpg</side>
  <side value="2">side2.jpg</side>
  <side value="3">side3.jpg</side>
  <side value="4">side4.jpg</side>
  <side value="5">side5.jpg</side>
  <side value="6">side6.jpg</side>
</object>
```

Objekte vom Typ **dice** haben als Werte Seitenobjekte mit Tag *side*, mit einem Attribut **value** und als Wert das Bild der Seite.



## **2 Spielsteuerung**