

# Bundeskampf-Bot

## *Dokumentation*

16. März 2011

### Inhaltsverzeichnis

<b>1</b>	<b>Runterladen</b>	<b>1</b>
<b>2</b>	<b>Schnellstart Windows</b>	<b>1</b>
<b>3</b>	<b>Allgemein</b>	<b>2</b>
<b>4</b>	<b>Ausgaben</b>	<b>2</b>
<b>5</b>	<b>Konfigurationsdatei</b>	<b>2</b>
<b>6</b>	<b>Programm-Ablaufplan</b>	<b>4</b>
6.1	Außendienst . . . . .	4
6.2	Arbeiten . . . . .	5
6.3	Bank . . . . .	5
6.4	Böse Spieler . . . . .	5
6.5	Angriff . . . . .	5
6.6	Kampf . . . . .	6
6.7	Minuten . . . . .	6
6.8	Stopp . . . . .	7
6.9	Beschreibung . . . . .	7
<b>7</b>	<b>Programm-Varianten</b>	<b>8</b>
7.1	Tagesquiz . . . . .	8
7.2	Rubbelos . . . . .	8
7.3	Lottoschein . . . . .	9
<b>8</b>	<b>Unerwartete Fehler</b>	<b>9</b>
8.1	Bekannte Fehler . . . . .	9
<b>9</b>	<b>Beispielkonfigurationen</b>	<b>10</b>

# 1 Runterladen

Auf der Webseite<sup>1</sup> kann man verschiedene Dateien runterladen. Java<sup>2</sup> wird für die Ausführung benötigt. Für einen einfachen Betrieb braucht man dann nur zwei Dateien:

- Bundeskampf.jar
- config.json

Die `.jar`-Datei heißt auf der Webseite, wie die aktuelle Version. Zur Zeit der Dokumentation lautet der Name "`1.0.beta5.jar`". Man kann diesen Namen umbenennen. Beide Dateien müssen im gleichen Verzeichnis liegen.

Die GZip-Datei mit dem Verzeichnis `src` beinhaltet den Quellcode. Diesen braucht man nur, falls man das Programm verändern möchte. Dazu sollte man beachten, dass der Code unter GPL<sup>3</sup> steht.

# 2 Schnellstart Windows

Da schon viele Windowsnutzer beim ersten Starten scheitern, habe ich eine Zip-Datei mit allen nötigen Dateien erstellt (`schnellstartWindows.zip`). Dort ist auch eine Stapelverarbeitungsdatei (`start.bat`) enthalten. Nach dem Entpacken einfach einmal diese Datei starten. Es sollte sich ein Fenster öffnen, welches erst nach Bestätigung wieder schließt. Nach dem ersten Starten sollte die Ausgabe ungefähr so aussehen:

```
Login  
Login failed
```

Der Login kann natürlich nicht stattfinden. Es fehlen die Logindaten in der Konfiguration. Diese ist mit Wordpad zu öffnen und folgend zu speichern:

Datei → Sichern unter → Dateityp: Unicode-Textdokument<sup>4</sup>

Sollte nun ein Login möglich sein, erklärt der Rest der Dokumentation, wie man den Bot einstellt. Sollte es nicht funktionieren, bitte auch zuerst dieses PDF lesen und dann beschweren.

<sup>1</sup><http://www.georf.de/bundeskampf/>

<sup>2</sup><http://www.java.com/de/download/>

<sup>3</sup><http://www.gnu.org/licenses/gpl.html>

<sup>4</sup><http://www.gutefrage.net/frage/wie-kann-ich-utf-8-speichern>

### 3 Allgemein

Das Programm arbeitet über die Kommandozeile. Öffnen Sie also zuerst eine Konsole und navigieren Sie in das Verzeichnis, wohin Sie die Datei geladen haben.<sup>56</sup>

Der allgemeine Aufruf dann sieht wie folgt aus:

```
eingabe# java -jar Bundeskampf.jar
```

Ruft man das Programm nur mit der Hilfe auf, erscheint folgendes:

```
eingabe# java -jar Bundeskampf.jar --help
Bundeskampf-Bot - Version: 1.0.beta4

Aufruf: Bundeskampf.jar [Optionen]
        Bundeskampf.jar [Optionen] quiz
        Bundeskampf.jar [Optionen] los
        Bundeskampf.jar [Optionen] lotto

Optionen:
  --help -h      Zeigt diese Hilfe an
  --config=FILE  Nimmt FILE als Konfiguration

Weitere Hinweise auf http://www.georf.de/bundeskampf
```

### 4 Ausgaben

Das Programm erzeugt Konsolenausgaben. Dabei schreibt es in stderr (Fehlerausgabe) und stdout (Normale Ausgabe). Es gibt nun Möglichkeiten diese Ausgaben getrennt, bzw. zusammen in eine Datei umzuleiten. Dadurch kann man mehrere Stunden zurückblicken. Die Konsole speichert nur eine bestimmte Anzahl von Zeilen, die man zurück scrollen kann. Die Ausgabe ist in Windows<sup>7</sup> und Linux<sup>8</sup> möglich.

### 5 Konfigurationsdatei

Normalerweise liegt die Datei `config.json` im gleichen Verzeichnis, wie das Programm. Die Codierung ist UTF-8 (Wichtig: UTF-8 **ohne BOM**). Es müssen bestimmte Angaben gemacht werden, ohne die das Programm beim Start

<sup>5</sup>Windows: [http://www.carpelibrum.de/tutorials/windows\\_konsole.pdf](http://www.carpelibrum.de/tutorials/windows_konsole.pdf)

<sup>6</sup>Linux: <http://wiki.ubuntuusers.de/Terminal>

<sup>7</sup><http://support.microsoft.com/kb/110930/de>

<sup>8</sup><http://www.fibel.org/linux/lfo-0.6.0/node111.html>

einen Fehler ausgibt. Dazu gehören: "Benutzername", "Passwort", "Ausgabe", "Außendienst verfügbar", "Kein Außendienst". Name und Passworter werden in ‘’ eingeschlossen und bezeichnen die Zugangsdaten. "Ausgabe" bezeichnet die Menge der Ausgabe. 0 → nur Fehler; 1 → Hinweise auf aktuelle Funktionen; 2 → alle Ausgaben. Die beiden weiteren Angaben werden als Listen bezeichnet und beinhalten die Funktionen, die im Abschnitt 6 erklärt werden.

Alle Zeilen, die mit der Raute (#) anfangen, werden nicht benutzt. Sie können also als Kommentare genutzt werden.

Beispiele für Konfigurationsdateien finden Sie im Abschnitt 9.

Hier folgt eine minimale Konfigurationsdatei.

```
"Benutzername": "name",
"Passwort": "geheim",

"Ausgabe": 2,

"Außendienst_verfügbar": [
],
"Kein_Außendienst": [
]
```

Name	Beschreibung	Pflicht	Datentyp
"Benutzername"		ja	String
"Passwort"		ja	String
"Ausgabe"	Wert für die Menge der Ausgabe	ja	Int
"Außendienst verfügbar"	Plan, wenn der Außendienst verfügbar ist	ja	Array
"Kein Außendienst"	Plan, wenn der Außendienst nicht verfügbar ist	ja	Array
"Angriff nochmal"	Wenn der Wert beim Angriff überschritten wird, wird der Gegner nochmal angegriffen	nein	Int
"Info Pfad"	Verzeichnis, in dem erweiterte Kampfberichte gespeichert werden sollen. Wird es weggelassen, werden keine gespeichert.	nein	String

## 6 Programm-Ablaufplan

Die eigentliche Aufgabe des Programms beinhaltet das Abarbeiten eines Plans nach bestimmten Kriterien. Dieser Plan wird in der Konfigurationsdatei (`config.json`) definiert. Es gibt zwei unterschiedliche Pläne, in denen einzelne Funktionen aufgerufen werden.

Der Plan 0, auch als "Außendienst verfügbar" bezeichnet, wird ausgeführt, falls man am aktuellen Tag noch einen Außendienst machen darf. Nach jedem Aufruf einer Funktion wird erneut geprüft ob noch Außendienste gemacht werden dürfen. Ist das nicht der Fall, wird in den Plan 1 übergegangen.

Der Plan 1, auch als "Kein Außendienst" bezeichnet, wird in zwei Fällen ausgeführt. Zum einen, wenn im Plan 0 keine Funktionen definiert sind, der Plan also leer ist. Zum anderen, wenn es für den aktuellen Plan keine Außendienste mehr ausgeführt werden können.

Die Pläne werden in der Config in einer Aufzählung beschrieben. Zwischen ihnen müssen also immer Kommata stehen. Hier ein Beispiel:

```
"Außendienst_verfügbar": [
    {"Außendienst": 0},
    {"Außendienst": 1},
    {"Bank": 0},
    {"Bank": 103},
    {"Böse": 0}
],
```

Wie man erkennen kann, werden die Funktionen in eckigen Klammern eingeschlossen. Nach der letzten Funktion kommt kein Kommata.

Nachfolgend werde alle implementierten Funktionen beschrieben. Sie können in beiden Plänen benutzt werden.

### 6.1 Außendienst

Der Befehl "Außendienst" ist nur in Plan 0 sinnvoll, kann also auch nur dort eingesetzt werden. Als Parameter kann 0 für leichten und 1 für schweren Außendienst gesetzt werden.

```
"Außendienst_verfügbar": [
    {"Außendienst": 0},
    {"Außendienst": 1}
],
```

## 6.2 Arbeiten

Der Befehl "Arbeiten" nimmt eine Zahl als Parameter. Diese stellt die Anzahl der Stunden ein.

```
"Außendienst_verfügbar":[
    {"Arbeiten":1}
],
```

## 6.3 Bank

Der Befehl "Bank" bringt Geld auf die Sparkasse. Mit dem Parameter 0 bringt man so viel wie möglich auf die Bank. Mit allen anderen positiven Zahlen wird genauso viel Geld eingezahlt, welches von der Zahl repräsentiert wird.

```
"Außendienst_verfügbar":[
# Bringt alles auf die Bank
    {"Bank":0},
# Bringt nur 205 DM auf die Bank
    {"Bank":205}
],
```

## 6.4 Böse Spieler

Der Befehl "Böse" greift eine Person aus der "Böse Spieler"-Liste an. Wählt man als Parameter 0, wird zufällig jemand angegriffen. Wird 1 gewählt, werden alle aus der Liste gleichmäßig angegriffen. Pro Befehl wird einmal angegriffen.

```
"Außendienst_verfügbar":[
# Böse-Liste-Person angreifen (zufällig = 0)
    {"Böse":0},
# Böse-Liste-Person angreifen (von oben = 1)
    {"Böse":1}
],
```

## 6.5 Angriff

Der Befehl "Angriff" bringt den Bot dazu sich einen geeigneten Gegner zu suchen und ihn anzugreifen. Dabei kann man zwei Parameter einstellen. "Stufe" bezeichnet die Anzahl der Level, die der Gegner Unterschied haben soll. Ist man selber im Level 8 und will jemanden aus Level 5 angreifen trägt man also -3 ein. Der zweite Parameter sagt was über die Vereinszugehörigkeit aus: -1 = Kein Verein; 0 = Egal; 1 = Im Verein.

```

"Außendienst_verfügbar":[
# Angriff mit eigene Stufe+1 und im Verein
  {"Angriff":{"Stufe":1,"Verein":1}},
# Angriff mit eigene Stufe-5 und nicht im Verein
  {"Angriff":{"Stufe":-5,"Verein":-1}},
# Angriff mit eigene Stufe und egal ob im Verein
  {"Angriff":{"Stufe":0,"Verein":0}},
],

```

## 6.6 Kampf

Der Befehl "Kampf" erzeugt im Gegensatz zu "Böse" und "Angriff" eine Keilerei mit einem bestimmten Gegner. Als Parameter wird entweder die ID oder der Name des Gegners übergeben. Die ID ist das Ende der URL, wenn man auf dem Profil ist. Bei der URL des Profils von "\_wolf\_" (1. Platz) lautet:

<http://www.bundeskampf.com/characters/profile/CHEGGF>

Die ID lautet also: CHEGGF Man könnte auch den Namen angeben, allerdings wird der bei zu kurzen Namen nicht immer gefunden. Hier beide Beispiele:

```

"Außendienst_verfügbar":[
  {"Kampf": "CHEGGF"},
  {"Kampf": "_wolf_"}
],

```

## 6.7 Minuten

Der Befehl "Minuten" bewirkt ein Schlafen des Bots. Als Parameter nimmt er die Anzahl der Minuten, die er Schlafen soll. Dieser Befehl muss nicht für die anderen Funktionen benutzt werden, sondern ist dafür da, dass man selber etwas machen kann.

```

"Außendienst_verfügbar":[
# Warte 3 Minuten
  {"Minuten":3},
# Warte 2 Stunden
  {"Minuten":120}
],

```

## 6.8 Stopp

Normalerweise arbeitet der Bot die Listen nacheinander ab und fängt wieder von vorne an. Um dies zu Unterbrechen gibt es den Befehl "Stopp".

```
"Außendienst_␣verfügbar": [
    {"Stopp": true}
],
```

## 6.9 Beschreibung

Diese Funktion ermöglicht es, den Bot zu steuern ohne ihn auszuschalten oder am richtigen Rechner zu sitzen. Man trägt die Funktion einfach in den Plan ein.

```
"Außendienst_␣verfügbar": [
    {"Beschreibung": true}
],
```

Immer wenn der Plan zu der Funktion "Beschreibung" kommt, sucht das Programm auf der eigenen Profilseite in der Beschreibung nach einem Befehl. Dieser muss folgende Form haben:

```
{"Befehl": [
]}
```

Zwischen den eckigen Klammern können beliebige Befehle stehen, wie in den anderen Plänen auch. Wichtig, dass das erste Zeichen die geschweifte Klammer ist. Der Bot schreibt dann Hinweise in diese Beschreibung, wenn er fertig ist oder die Eingabe nicht richtig verstehen konnte.



Seit der Version **beta5** gibt es eine Erweiterung hierzu. Der Platz in der Beschreibung ist begrenzt, also kann man Angriffe in einer einfachen Liste ausfüh-



ren. Der folgende Befehl so direkt in die Beschreibung eingegeben, würde zwei Angriffe auf "\_wolf\_" ausführen.

```
["CHEGGF", "_wolf_"]
```

## 7 Programm-Varianten

Das Programm implementiert verschiedene Hilfsmittel. Hier werden Sie aufgeführt und beschrieben.

### 7.1 Tagesquiz

*Bei unserem Tagesquiz hast Du jeden Tag die Chance D-Mark oder Fortschritt zu gewinnen. Für jede richtige Frage bekommst Du je nach beliebigen D-Mark oder Fortschritt gutgeschrieben. Je höher deine Stufe, desto höher wird auch die Belohnung. Hast Du mindestens 7 von den 10 Fragen richtig beantwortet bekommst Du sogar 20% Belohnung extra! Und nun viel Spaß!*

Mit Hilfe des Programms kann das Tagesquiz ziemlich gut gelöst werden. Und es geht viel schneller, als wenn man es selber macht. Der Aufruf sieht wie folgt aus:

```
eingabe# java -jar Bundeskampf.jar quiz
```

Beispielausgabe:

```
Login
visit: http://www.bundeskampf.com/quiz/
Get questions:
Eine dumme Frage, die es gar nicht gibt?
Found in database: Antwort CORRECT
Eine Frage, die noch nie gestellt wurde?
Google say: findet Google CORRECT
[.....]
Quiz: get 1140 money, with 10 correct answers
Logout
visit: http://www.bundeskampf.com/signups/logout
```

Dabei wird zuerst in einer Datenbank nach der Antwort gesucht. Falls die Frage dort nicht gefunden wird, wird Google nach einer passenden Antwort befragt.

### 7.2 Rubbelos

Wer schon mal selber versucht hat, die Rubbellose von ihrer Beschichtung zu befreien, der lässt es sehr schnell sein, denn es dauert zu lange. Will man es schneller erledigen, nimmt man folgenden Befehl:

```
eingabe# java -jar Bundeskampf.jar los
```

Beispielausgabe:

```

Login
visit: http://www.bundeskampf.com/kiosk/
visit: http://www.bundeskampf.com/kiosk/losKaufen/
visit: http://www.bundeskampf.com/kiosk/lose/
Lot: WIN D-Mark: 50
visit: http://www.bundeskampf.com/kiosk/getLosWin
visit: http://www.bundeskampf.com/kiosk/
visit: http://www.bundeskampf.com/kiosk/losKaufen/
visit: http://www.bundeskampf.com/kiosk/lose/
Lot: WIN D-Mark: 50
visit: http://www.bundeskampf.com/kiosk/getLosWin
visit: http://www.bundeskampf.com/kiosk/
visit: http://www.bundeskampf.com/kiosk/losKaufen/
visit: http://www.bundeskampf.com/kiosk/lose/
Lot: WIN D-Mark: 50
visit: http://www.bundeskampf.com/kiosk/getLosWin
Logout
visit: http://www.bundeskampf.com/signups/logout

```

### 7.3 Lottoschein

*Genau wie im Fernsehen könnt ihr es euch Sonntags Abend gemütlich machen und darauf hoffen, dass die Glücksfee euch diese Woche hold ist. Und ihr - zumindest in Bundeskampf - nicht mehr zu arbeiten braucht.*

Das Ausfüllen eines Scheins dauert auch ein wenig. Und welche Zahlen soll man nehmen? Mit dem folgenden Aufruf erledigt es das Programm:

```
eingabe# java -jar Bundeskampf.jar lotto
```

Beispielausgabe (die Zahlen werden zufällig gewählt):

```

Login
visit: http://www.bundeskampf.com/lotto/history
Added numbers: 2,5,10,38,40
Logout
visit: http://www.bundeskampf.com/signups/logout

```

## 8 Unerwartete Fehler

Sollte ein unerwarteter Fehler auftreten, welches zu einem Absturz des Systems führt, wird der Fehler mit mehreren Hinweisen ausgegeben. Danach startet sich das Programm neu. Das Programm versucht dann erneut seine Aufgaben zu erledigen. Passiert zu oft ein Absturz beendet sich das Programm.

Sollten Sie sich den Fehler durch die Ausgaben des Bots und durch selbstständigen Prüfen auf der Webseite nicht erklären lassen, senden Sie die Ausgaben des Bots per E-Mail an den Entwickler.

### 8.1 Bekannte Fehler

Tritt ein Fehler auf, in dem von einem *Internal Server Error - 500* die Rede ist, liegt es an dem Bundeskampf-Server und nicht am Bot. Die genaue Fehlermeldung ist auf der Webseite zu finden.

## 9 Beispielkonfigurationen

Die folgenden Konfigurationen sind auch auf der Webseite zum Download verfügbar.

Eine Konfiguration für jemanden, der nur die Außendienste (schwer) machen will und danach das Programm beenden will, sieht dann so aus:

```
"Benutzername": "meinName",
"Passwort": "geheim",
"Ausgabe": 2,
"Außendienst_verfügbar": [
    {"Außendienst": 1}
],
"Kein_Außendienst": [
    {"Stopp": true}
]
```

Konfiguration: Nur Außendienste leicht, wenn fertig, regelmäßig prüfen, ob es schon ein neuer Tag ist und dann wieder Außendienste machen:

```
"Benutzername": "meinName",
"Passwort": "geheim",
"Ausgabe": 2,
"Außendienst_verfügbar": [
    {"Außendienst": 0}
],
"Kein_Außendienst": [
    {"Minuten": 30}
]
```

Konfiguration: Keine Außendienste; Nur Angreifen, gleiche Stufe, im Verein.

```
"Benutzername": "meinName",
"Passwort": "geheim",
"Ausgabe": 2,
"Außendienst_verfügbar": [
],
"Kein_Außendienst": [
    {"Angriff": {"Stufe": 0, "Verein": 1}}
]
```

Konfiguration: Keine Außendienste; Nur Arbeiten, dann alles auf Bank bringen.

```
"Benutzername": "meinName",
"Passwort": "geheim",
"Ausgabe": 2,
"Außendienst_verfügbar": [
],
"Kein_Außendienst": [
```

```
    {"Arbeiten":1},  
    {"Bank":0}  
]
```

Konfiguration: Keine Außendienste; Angreifen, dann einen bösen Angreifen und dann eine Pause.

```
"Benutzername":"meinName",  
"Passwort":"geheim",  
"Ausgabe":2,  
"Außendienst_verfügbar":[  
],  
"Kein_Außendienst":[  
    {"Angriff":{"Stufe":0,"Verein":1}},  
    {"Böse":1},  
    {"Minuten":10}  
]
```



## Commons Deed

Namensnennung - Keine kommerzielle Nutzung - Weitergabe unter gleichen Bedingungen 3.0

Es ist Ihnen gestattet:

- das Werk vervielfältigen, verbreiten und öffentlich zugänglich machen
- Abwandlungen bzw. Bearbeitungen des Inhaltes anfertigen

Zu den folgenden Bedingungen:

- *Namensnennung.* Sie müssen den Namen des Autors/Rechtsinhabers in der von ihm festgelegten Weise nennen.
- *Keine kommerzielle Nutzung.* Dieser Inhalt darf nicht für kommerzielle Zwecke verwendet werden.
- *Weitergabe unter gleichen Bedingungen.* Wenn Sie den lizenzierten Inhalt bearbeiten oder in anderer Weise umgestalten, verändern oder als Grundlage für einen anderen Inhalt verwenden, dürfen Sie den neu entstandenen Inhalt nur unter Verwendung von Lizenzbedingungen weitergeben, die mit denen dieses Lizenzvertrages identisch oder vergleichbar sind.
- Im Falle einer Verbreitung müssen Sie anderen die Lizenzbedingungen, unter die dieser Inhalt fällt, mitteilen.
- Jede der vorgenannten Bedingungen kann aufgehoben werden, sofern Sie die Einwilligung des Rechteinhabers dazu erhalten.
- Diese Lizenz lässt die Urheberpersönlichkeitsrechte unberührt.

Das Commons Deed ist eine Zusammenfassung des Lizenzvertrags in allgemeinverständlicher Sprache. Um den Lizenzvertrag einzusehen, besuchen Sie die Seite

<http://creativecommons.org/licenses/by-nc-sa/3.0/de/>

oder senden Sie einen Brief an Creative Commons, 543 Howard Street, 5th Floor, San Francisco, California, 94105, USA.

