
Social Bots

— Eine Unterrichtseinheit zur
Netzwerkkommunikation —

Benjamin Knorr, 05.07.2019

Informatikstunde - Lerninhalte

- Social Bots:

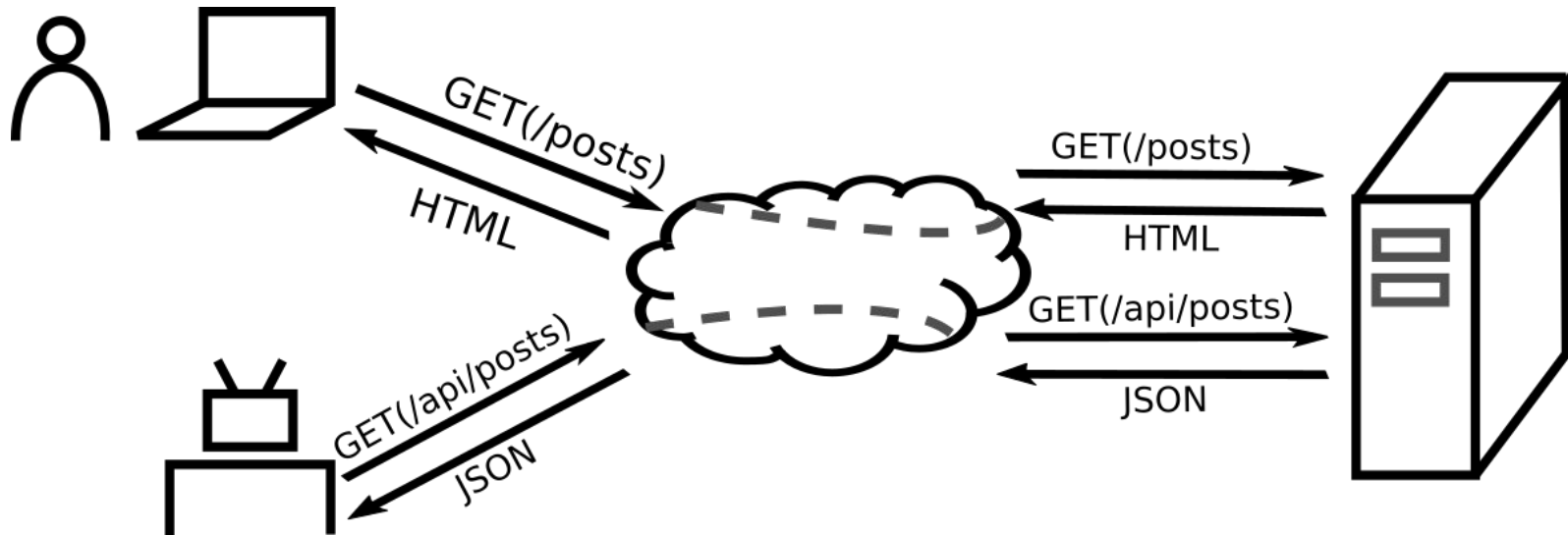
 - Sensibilisierung für Existenz

 - Verständnis der Funktionsweise auf konzeptueller Ebene

Erstellen eines Social Bots zeigt Funktionsweise auf → Dadurch Gefahren und Einschränkungen abschätzbar!

Informatikstunde - Lerninhalte

- Kommunikation mit Webservern:
GET und POST Anfragen des Protokolls HTTP
Datenformat JSON





Öffentliche Pinnwand

Was denkst du gerade, root?

Schreibe eine Nachricht...



Meiste Likes

Sortieren nach: [Likes](#), [Datum](#)



FabiDev

Hello World

— Mar 19, 2019 7:57:11 PM

Gefällt mir Gefällt: [FabiDev](#) , [danielgo](#) , [Hello](#) und 3 weiteren.



root

Hallo, Welt!

— Mar 17, 2019 7:54:48 PM

Gefällt mir Gefällt: [Hello](#) , [Wetterfuehlig](#) , [HiHiHi](#) und 2 weiteren.

Öffentliche

Was denkst du gerade, root?

Schreibe eine Nachricht...

Meiste Likes



Hello World

— Mar 19, 2019 7:57:11 PM

Gefällt mir Gefällt: FabiDev , danielgo , Hello und 3 weiteren.



Hallo, Welt!

— Mar 17, 2019 7:54:48 PM

Gefällt mir Gefällt: Hello , Wetterfuehlig , HiHiHi und 2 weiteren.

https://www.socialbotnet.de/api/posts

JSON

Rohdaten

Kopfzeilen

```
id: 1
message: "Hallo, Welt!"
▼ user:
  id: 1
  username: "root"
  hobbies: "Netzwerken"
  about: "Ich bin root"
publishingDate: "2019-03-17 19:54:48"
▼ likedBy:
  ▼ 0:
    id: 3
    username: "Hello"
    hobbies: "Grüßen"
    about: "Ich grüße jeden Nutzer"
  ▼ 1:
    id: 21
    username: "N0name123"
```

Ausprobieren

1. Erstellen sie einen Benutzer auf www.socialbotnet.de und probieren Sie ein paar Funktionen des Netzwerks im Browser aus.
2. Laden Sie die Projektvorlage "Projekt mit Objekten" von der Materialseite herunter.

Kurze Einführung in die Technik: POST

POST-Anfragen: Senden von Daten an den Server.

```
public void posten(String nachricht) {  
    NetzwerkZugriff socialbotnet = new NetzwerkZugriff("http://www.socialbotnet.de");  
    socialbotnet.POSTAnfrageVorbereiten("username", username);  
    socialbotnet.POSTAnfrageVorbereiten("password", password);  
    socialbotnet.POSTAnfrageVorbereiten("message", nachricht);  
    socialbotnet.POSTAnfrageSenden("/api/post");  
}
```

Anmerkung: Code-Vorlage von <https://www.socialbotnet.de/material> mit Hilfsklasse "NetzwerkZugriff" für Kommunikation mit Web-Servern bereitstellt.

Netzwerkzugriff
NetzwerkZugriff(String domain) GETAnfrageSenden(String url) : String POSTAnfrageVorbereiten(String parameterName, String parameterWert): void POSTAnfrageSenden(String url) : void

Ausprobieren

1. Erstellen sie einen Benutzer auf www.socialbotnet.de und probieren Sie ein paar Funktionen des Netzwerks im Browser aus.
2. Laden Sie die Projektvorlage "Projekt mit Objekten" von der Materialseite herunter.
3. Probieren Sie die POST-Schnittstellen im Projekt aus, um Beiträge zu schreiben oder zu liken.
(Falls nötig: Sie können die GET- Schnittstellen im Browser aufrufen, um eine Datenansicht zu erhalten um z.B. ids zu sehen)

Kurze Einführung in die Technik: GET (1/2)

GET-Anfragen: Abrufen von Daten vom Server.

```
public void eigenePinnwandLiken() {  
    // Alle Posts auf der eigenen Pinnwand abrufen  
    String antwort = socialbotnet.GETAnfrageSenden("/api/pinnwand/"+this.username);  
    // Mit dem vorgefertigten JSON-Parser zu gewohnten Objekten umwandeln.  
    Post[] posts = AntwortParser.zuPostArray(antwort);  
    // Iteration über die einzelnen Posts  
    for (int i=0; i<posts.length; i++) {  
        Post post = posts[i];  
  
        // Von dem Post wird die ID als Parameter zum Liken benötigt  
        int id = post.getId();  
        liken(id);  
    }  
}
```

Kurze Einführung in die Technik: GET (2/2)

GET-Anfragen: Abrufen von Daten vom Server.

Mit den verarbeiteten Daten können dann neue POST-Anfragen erstellt werden.

```
public void liken(int id) {  
    socialbotnet.POSTAnfrageVorbereiten("username", this.username);  
    socialbotnet.POSTAnfrageVorbereiten("password", this.password);  
    socialbotnet.POSTAnfrageVorbereiten("postid", id);  
    socialbotnet.POSTAnfrageSenden("/api/like");  
}
```

Ausprobieren

1. Erstellen sie einen Benutzer auf www.socialbotnet.de und probieren Sie ein paar Funktionen des Netzwerks im Browser aus.
2. Laden Sie die Projektvorlage “Projekt mit Objekten” von der Materialseite herunter.
3. Probieren Sie die POST-Schnittstellen im Projekt aus, um Beiträge zu schreiben oder zu liken.
(Falls nötig: Sie können die GET- Schnittstellen im Browser aufrufen, um eine Datenansicht zu erhalten um z.B. ids zu sehen)
4. Verarbeiten Sie die Daten einer GET-Anfrage

Fortgeschrittene Anwendungen

Andere Schnittstellen des Servers

Browser verwenden gleiches Protokoll (HTTP) => Erkenntnis: Auch mit den Browser-Schnittstellen kann genau so kommuniziert werden!

Beispiel: POST-Anfrage an <https://www.socialbotnet.de/registrieren>, mit Registrierungs-Daten wie im Browser -> Ermöglicht automatisches Erstellen von Bots.

Schutzmöglichkeiten: z.B. CAPTCHAs (bei anderen Webseiten üblich)

Fortgeschrittene Anwendungen

Andere JSON-Webseiten

Mit der Verarbeitung von JSON-Daten sind zahlreiche Webseiten anbindbar

Beispiel:

<https://openweathermap.org/api> stellt aktuelles Wetter per JSON-API zur Verfügung.

```
public double temperaturAbrufen() {  
    NetzwerkZugriff openWeather = new NetzwerkZugriff(  
        "https://api.openweathermap.org/data/2.5"  
    );  
    String antwort = openWeather.GETAnfrageSenden(  
        "/weather?q=munich&units=metric&appid=" + appid  
    );  
    JSONObject antwortJSON = new JSONObject(antwort);  
    JSONObject main = antwortJSON.getJSONObject("main");  
    double temperatur = main.getDouble("temp");  
    return temperatur;  
}
```

Material

Webseite: <https://www.socialbotnet.de>

Projektvorlage und Handouts zur selbstständigen Erarbeitung:
<https://www.socialbotnet.de/material>

Alles Weitere:

- Weiterentwicklung des Codes auf Github (-> Siehe Link in Footer, Feedback & Bugreports herzlich willkommen!)
- .jar Datei für lokales SocialBotNet ebenfalls auf Github verfügbar.

Vielen Dank für Ihre Aufmerksamkeit

Ich freue mich über Feedback, Anregungen und Fragen

E-Mail Kontakt: knorr.b@gmx.de

Anhang: JSON-Format

Zwei Grundstrukturen: Objekte mit Key-Value Paaren und Arrays aus Objekten

Objekte:

```
{  
  "key ": "value",  
  "key2 ": 42,  
  "key3 ": true ,  
  ...  
}
```

Arrays:

```
[  
  {  
    "key ": "objekt1",  
  },  
  {  
    "key ": "objekt2"  
  },  
  ...  
]
```


Anhang: JSON in Java

```
public void neuestenBeitragKopieren() {  
    NetzwerkZugriff socialbotnet = new NetzwerkZugriff("http://www.socialbotnet.de");  
    String antwort = socialbotnet.GETAnfrageSenden("/api/posts");  
    // Antwort in JSON-Array umwandeln  
    JSONArray array = new JSONArray(antwort);  
    // Erstes Objekt aus dem Array  
    JSONObject neusteNachricht = array.getJSONObject(0);  
    // String mit dem Key "message" holen  
    String nachricht = neusteNachricht.getString("message");  
  
    posten(nachricht);  
}
```