

# Projektplan

## Beschreibung:

Das Ziel des Projekts ist die Erstellung eines Programms, welches Daten aus einem Youtube Kanal ausliest, speichert und ausgibt. Hierfür soll nur der Link zu dem Kanal in den Code eingefügt werden und das Programm demnach für unterschiedliche Kanäle angewendet werden können. Als Informationen sollen zunächst der Name des Kanals und die Anzahl der Abonnenten ausgelesen werden. Danach soll die Anwendung alle Videos, die auf dem Kanal hochgeladen wurden, auswerten. Bei den einzelnen Videos sollen zunächst die momentane Viewerszahl und die verwendeten Tags verwertet werden. Diese Informationen werden einzeln gespeichert und am Ende zusammengeführt. Es werden die durchschnittliche Viewerszahl berechnet und die Tags nach Häufigkeit gelistet. Diese werden im letzten Schritt zusammen mit den grundsätzlichen Kanalinformationen, wie der Name und die Anzahl der Abonnenten ausgegeben. Optional könnten diese mit weiteren Daten wie häufig genutzten Wörtern im Titel, durchschnittliche Länge des Videos, oder durchschnittliche Anzahl der Kommentare ergänzt oder z.B. CSV Datei exportiert werden.

Ebenfalls möglich ist eine praktische Verwendung des Programms mit dem Vergleich mehrerer thematisch ähnlicher Youtube-Channel denkbar. Das Programm würde mit mehreren Kanälen durchlaufen und die Ergebnisse miteinander verglichen werden. Mit der Anzahl der Viewer und den verwendeten Tags könnten interessante Schlussfolgerungen gezogen werden. Hierbei gibt es allerdings mehrere Punkte zu beachten. Zunächst gäbe es viele Tags, die dem Namen des gemeinsamen Themas entsprechen, und nicht ausgewertet werden können. Zusätzlich fehlen viele wichtige Informationen, die einen großen Teil zu dem Erfolg von Videos beitragen. Thumbnails und Verlinkungen stellen einen sehr großen Teil dar, der keine Beachtung finden würde. Dies würde zweifelsfrei zu einer nur bedingten Aussagekraft der Schlussfolgerungen führen.

Eine weitere Möglichkeit der Weiterverarbeitung stellt die Darstellung der Ergebnisse in einem GUI dar. In diesem würden die grundsätzlichen Kanalinformationen in der Kopfzeile wiedergegeben werden. Die am meisten verwendeten 15 Tags könnten darunter graphisch mit Säulendiagrammen dargestellt werden. Darunter könnte ein "Zurück" Button eine weitere Durchführung des Ablaufs mit einem anderen Youtube Kanal ermöglichen (s. Abb. 1).

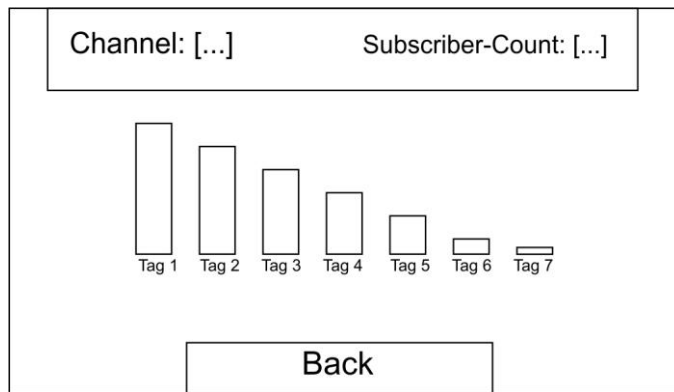


Abb. 1: optionales GUI mit Ergebnissen

### Verwendete Programme und Tools:

Als Programmiersprache wird ausschließlich Python 2.7 verwendet. Die Bibliotheken BeautifulSoup und HTML Request dienen zur Auswertung von den erforderlichen Websitedaten im HTML Code und zur Erstellung des Crawlers. Die benötigten Informationen wie Anzahl der Views oder Tags sind über den Quelltext der Website, in der das Video angezeigt wird, auslesbar. Alternativ sind Ergänzungen durch den Google Youtube API Feed möglich. Hier ergeben sich aber die Probleme mangelnder Orientierungsmöglichkeiten und Vorlagen z.B. von einem Crawler durch Youtube, weshalb dies nur als Alternative beim Auftreten von größeren Fehlern oder fehlender Kompatibilität dient.

Als Vorlage für die Auswertung von Videos eines Channels dient ein Code, welcher alle Videos eines Kanals namentlich mit entsprechender Anzahl von Views auflistet (s. Anhang). Dieser hat jedoch kleine formale Mängel und verwendet Tkinter zum Aufbau eines GUIs. Dennoch bietet er stark umgewandelt und angepasst eine gute Vorlage für einen Crawler durch einen Youtube Channel.

### Programmablauf:

Es folgt eine grobe Skizze der Organisation und des Ablaufs innerhalb des Programmcodes.

- Import BeautifulSoup, Requests etc.
- Input Youtube Link [Channel Seite]
- Auslese Name des Youtubers [im HTML Text unter <title>]
- Abspeichern Youtuber Name in Variable
- Auslese der Abonnentenzahl [im HTML Text unter „Subscriber Count“]
- Abspeichern Abonnentenzahl in Variable
- Implementierung Crawler
- Auslesung Tags [HTML unter keywords]

- Entfernung Trennzeichen [,] zwischen den Tags
- Einfügen der Tags in Liste
- Auslesung momentane Viewerzahl [HTML unter view\_count]
- Abspeichern Viewerzahl in Variable
- Addition der Viewerzahlen
- Einfügen aller Tags in Liste
- Zählen der Übereinstimmungen in der Liste der Tags
- Verbindung Tags mit Häufigkeit in einem Dictionary [ Tag - Anzahl Verwendungen]
- Sortierung nach Anzahl der Verwendungen
- Durchschnittliche Viewerzahl (Addition aller Views / Anzahl Videos) berechnen
- Ausgabe Kanalname [Zugriff auf Variable] mit entsprechendem Text [ Kanalname: ...]
- Ausgabe Abonnentenzahl [Zugriff auf Variable] mit entsprechendem Text [ Anzahl der Abonnenten: ...]
- Ausgabe Dictionary mit gelisteten verwendeten Tags

#### Anhang:

Link zu dem Quellcode des Youtube-Channel-Crawlers von "VISWESWARAN1998":

<http://codereview.stackexchange.com/questions/128936/the-youtube-crawler>

Youtube API Python:

[https://developers.google.com/youtube/1.0/developers\\_guide\\_python?csw=1](https://developers.google.com/youtube/1.0/developers_guide_python?csw=1)