Claudio Mondillo

ZÜrcher Lehrbetriebsverband ICT | CMONDILLO

Modul 426

Erkenntnisse

Inhaltsverzeichnis

[Einleitung 2](#_Toc89419124)

[Ursachen für Probleme 3](#_Toc89419125)

[Tag 1: 3](#_Toc89419126)

[Tag 2: 3](#_Toc89419127)

[Tag 3: 3](#_Toc89419128)

[Tag 4: 3](#_Toc89419129)

[Verbesserungsmassnahmen 3](#_Toc89419130)

[Tag 1: 3](#_Toc89419131)

[Tag 2: 3](#_Toc89419132)

[Tag 3: 3](#_Toc89419133)

[Tag 4: 3](#_Toc89419134)

[Gewonnene Erkenntnisse 3](#_Toc89419135)

[Tag 1: 3](#_Toc89419136)

[Tag 2: 4](#_Toc89419137)

[Tag 3: 4](#_Toc89419138)

[Tag 4: 4](#_Toc89419139)

# Einleitung

Ziel ist es eine Applikation zu erstellen, welche es ermöglicht die Follower Anzahl verschiedener Plattformen zu bekommen und diese dem User anzuzeigen. Dies ermöglicht dem Benutzer eine zentrale Stelle zu haben, welche alle Follower der verschiedenen Plattformen an einem Ort anzeigt, damit der Benutzer nicht mehrere Applikationen öffnen muss und schnell und einfach erkennt, ob seine Follower mehr oder weniger werden.

# Ursachen für Probleme

## Tag 1:

* Instagram und Facebook API sind sehr umständlich

## Tag 2:

* 7 verschiedene Programmiersprachen sind unnötig

## Tag 3:

* Daten werden im local storage gespeichert, -> besser in einem JSON abspeichern
* Promise macht probleme

## Tag 4:

* Funktion wird immer ausgeführt bevor sie soll. Deswegen ist es erforderlich, 2x den Button zu drücken, bis die Seite aktualisiert wird

# Verbesserungsmassnahmen

## Tag 1:

Instagram und Facebook API sind sehr umständlich: deshalb werden wir eine eigene API erstellen auf welcher wir alle Daten speichern und von dieser dann holen werden.

## Tag 2:

Antstatt 7 verschiedene Porgrammiersprachen zu verwenden haben wir uns geeinigt, eine Electron App zu erstellen. Dies erspart und den Server Teil mit dem erstellen einer eigenen API.

## Tag 3:

JSON Datei erstellen, welche die Daten speichert, anstatt local storage.

## Tag 4:

# Gewonnene Erkenntnisse

## Tag 1:

Am ersten Tag haben wir uns Gedanken gemacht, was wir genau machen wollen und mit welchen Programmiersprachen wir arbeiten.

Das Ziel ist eine Anwendung zu schreiben, welche die Follower von verschiedenen Plattformen zusammenzählt, damit der Benutzer einen schnellen Überblick über seine Follower bekommt und sofort sieht ob diese mehr oder weniger geworden sind.

Wir bekommen die Follower Anzahl über verschiedene Wege (API, HTML) und erstellen eine eigene API auf welcher wir die Follower pro Plattform speichern.

Mit Hilfe dieser API ist es uns dann möglich diese Zahlen auf unserer PHP Seite zu veranschaulichen.

**Neu gelerntes:**

-API erstellen mithilfe einer automatisch generierten db.json Datei und über Server laufen lassen

-Follower über Twitter API bekommen

-Replit kennengelernt

## Tag 2:

Den Weg, alles mithilfe einer Electron App zu visualisieren ist viel besser, da man dann nicht viele verschiedene Programmiersprachen benötigt und neben der Web App auch noch gleich eine Applikation für Android und Iphone erstellen kann.

**Neu gelerntes:**

-Wenige Programmiersprachen zu benutzen macht alles viel übersichtlicher und spätere Änderungen sind einfacher.

## Tag 3:

Wir werden eine JSON-Datei erstellen um die Daten zu speichern, da diese dann auch bei einem Storage Clean behaltet werden.

Das promise macht Probleme da nicht mit der API Abfrage gewartet wird, bis die Userdaten eingegeben wurden.

**Neu gelerntes:**

-Funktion von promise

## Tag 4:

Wir versuchten einen Fehler zu beheben, die Daten bei einem Klick auf den Button zu aktualisieren.

Probleme:

Yassin arbeitet am Frontend und Alex am Backend. Ich habe in dieser Zeit keine wichtigen Aufgaben, welche ich erledigen kann, weshalb ich mich damit beschäftige für die anderen Sozialen Medien schon heraus zu finden, wie ich dort die Follower bekomme. Ich lese mich in die Doc’s ein und bereite die Abfragen vor, welche wir später dann einbauen werden sobald die Hauptprobleme gelöst sind.