

Stockbird

Rebecca Kietzer
r.kietzer@oth-aw.de
OTH Amberg Weiden

Baran Baygin
b.baygin@oth-aw.de
OTH Amberg Weiden

Carl Küschall
c.kueschall@oth-aw.de
OTH Amberg Weiden

Jonathan Okorafor
j.okorafor@oth-aw.de
OTH Amberg Weiden

Luca Käsmann
l.kaesmann@oth-aw.de
OTH Amberg Weiden

Michael Zimmet
m.zimmet@oth-aw.de
OTH Amberg Weiden

Michael Ippisch
m.ippisch@oth-aw.de
OTH Amberg Weiden

Abstract—Dieses elektronische Dokument ist eine Vorlage. Die verschiedenen Komponenten Ihres Papiers [Titel, Text, Überschriften usw.] sind bereits auf dem Style Sheet definiert, wie in diesem Dokument dargestellt.

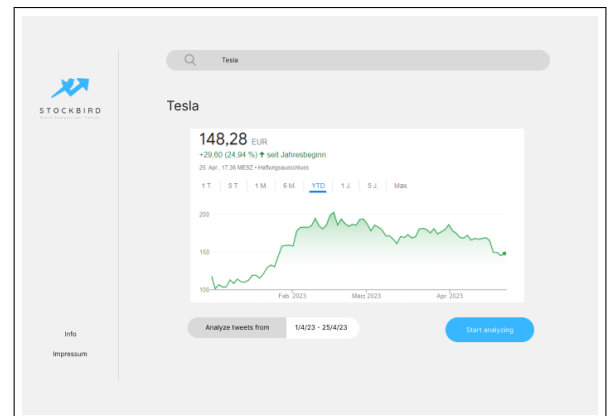
I. EINLEITUNG

StockBird ist eine innovative Webanwendung, die es ermöglicht den „unbekannten“ Einfluss der Tweets (Beiträge auf Twitter) wohlhabender Leute auf den Aktienmarkt aufzudecken und zu veranschaulichen. Bei dem 2006 unter dem Namen „Twtrr“ gegründeten Unternehmen handelt es sich um einen Microblogging Dienst, den man auch als öffentliches Tagebuch verstehen könnte. Hierbei wären die Tagebuchseiten die „Pages“ der einzelnen Nutzer, auf die jederzeit zugegriffen und denen gefolgt werden kann. Die Tagebucheinträge wären die „Tweets“ (kurze Texte bis 280 Zeichen), die wiederum andere Nutzer lesen, beantworten, oder aber auch „retweeten“ (erneutes tweeten oder zitieren anderer Tweets) können. Interessant sind so für viele Leute vor allem die Pages von bekannten Persönlichkeiten, die dort Ausschnitte aus ihrem Leben, oder aber auch Meinungen zu aktuellen Themen teilen. So kommt es, dass bekannte Unternehmer, wie beispielsweise Elon Musk eine Followerzahl von 137 Millionen Nutzern aufweisen können. So eine immense Reichweite bringt großen Einfluss mit sich. Besonders deutlich wird dies am Aktienmarkt, auf dem oben genannter schon des Öfteren Firmenkurse durch „beiläufige“ Tweets maßgeblich beeinflusst hat. StockBird liefert hier die Möglichkeit eben solche „Eingriffe“ in den Aktienmarkt zu erkennen, zu verstehen und zu veranschaulichen, ohne vorher unzählige Aktienartikel durchforsten zu müssen.

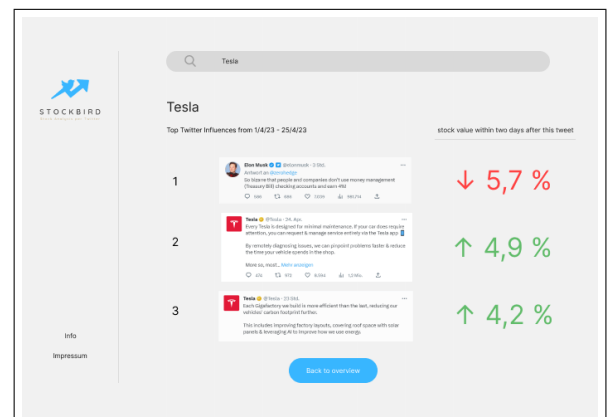
A. Motivation

Als Motivation diene zum einen ein von vornherein starkes Interesse am Aktienmarkt einiger Gruppenmitglieder und zum andern das Interesse an den technischen Innovationen Elon Musks (oder genauer seiner Firmen) der anderen. Dass Musk den Aktienmarkt mittels Tweets gelegentlich geringfügig beeinflusst, war naheliegend. Das Ausmaß, in dem dies

vonstattengeht, jedoch absolut nicht. Unser Interesse an der Kombination beider Themen war geweckt. Nun ist Musk allerdings nicht der einzige Unternehmer mit einem bekannten Twitter Account und bei weitem nicht die einzige Person, die Interesse daran hätte, den Aktienmarkt zu ihren Gunsten zu beeinflussen. Um dann die Wer? Wann? Wie stark? Fragen effizient beantworten zu können, ward StockBird geboren.



Auswählen und Eingrenzen der Aktien



Präsentation der einflussreichsten Tweets

II. Anforderungen

A. MUSS-Anforderungen

1) Stock [6] Auswahl

Ich als Nutzer möchte einen Stock auswählen können, um diesen genauer zu betrachten.

Akzeptanzkriterien:

- a) Eine Suchleiste soll zur Verfügung stehen
- b) Ein Dropdown Menü soll zur Verfügung stehen

2) Stock Visualisierung

Ich als Nutzer möchte den Stock graphisch betrachten können, um einen besseren Überblick über die Daten zu bekommen.

Akzeptanzkriterien:

- a) Nutzer kann heran- und herauszoomen
- b) Die Darstellung startet in einer herausgezoomten Ansicht

3) Betrachtungszeitraum Einschränken

Ich als Nutzer möchte den Betrachtungszeitraum für die Stock-Daten einschränken können, um den Fokus auf starke Schwankungen in den Daten zu setzen.

Akzeptanzkriterien:

- a) Zeitraum kann per Date-Picker, graphisch oder über zwei Eingabefelder eingeschränkt werden.
- b) Nach Eingabe passt sich die Visualisierung an.

B. SOLL-Anforderungen

1) Relevante Tweets einsehen

Ich als Nutzer möchte für meinen gewählten Stock relevante Tweets aus dem Beeinflussungszeitraum einsehen können, um Zeit zu sparen und somit schneller zu einem Ergebnis meiner Analyse zu kommen.

Akzeptanzkriterien:

- a) Die visualisierten Tweets beinhalten für den Stock relevante Hashtags und/oder Text jeglicher Art.
- b) Es wird Statistik für die Macht jedes relevanten Tweets berechnet und angezeigt.
- c) Die Tweets werden automatisch nach Statistik sortiert.
- d) Die Visualisierung der Daten des gewählten Stocks soll nach Analyse noch sichtbar sein

2) Zwei Ebenen der Analyse

Ich als Nutzer möchte zwischen den drei Ebenen 'Filtern', 'Analyse' auswählen können, um flexibel auf unterschiedliche Art analysieren zu können.

Akzeptanzkriterien:

- a) Nutzer kann über Menüpunkte zwischen den Ebenen wechseln.
- b) Auf der Ebene 'Filtern' kann der Nutzer die Tweets nach gewissen Attributen filtern
- c) Auf der Ebene 'Analyse' kann der Nutzer die Funktionen der Soll-Anforderung 'Relevante Tweets einsehen' anwenden.

C. OPTIONAL-Anforderungen

1) Sortierte Dropdownliste der Stocks

Ich als Nutzer möchte die Stocks in der Dropdown Liste sortiert und untergeordnet sehen.

Akzeptanzkriterien:

- a) Die Stocks werden strukturiert im 'Tree' Format in der Dropdownliste präsentiert.

2) Automatische Analyse

Ich als Nutzer möchte in einer weiteren dritten Ebene: Automatische Analyseinteressante Events/Spikes in den Daten meines gewählten Stocks anzeigen lassen, sodass ich diese auswählen kann und mir dafür Analyse wie in der Soll-Anforderung 'Relevante Tweets einsehen' durchgeführt wird.

Akzeptanzkriterien:

- a) Spikes werden automatisch berechnet und visualisiert.
- b) Nach Auswahl der Spikes wird die Analyse durchgeführt.
- c) Der Nutzer kann dynamisch zwischen den Spikes hin und her wechseln.

3) Bewertung der Tweets

Ich als Nutzer möchte, dass den relevanten Tweets ein Anteil zugeschrieben wird, welcher den Grad der Verantwortung für einen Spike zuteilt, um die Macht der Tweets zu erwägen.

Akzeptanzkriterien:

- a) Tweets erhalten einen Wert $\frac{x}{100}$
- b) Die Tweets werden mit ihrem jeweiligem Wert visualisiert.
- c) Die Tweets werden nach diesem Wert sortiert.

III. Technisches Grobkonzept

A. AWS S3 Bucket

Enthält Dateien der statischen Website[1]

B. AWS Lambda

Lambda Funktion fungiert als Backend für die API.[2]

C. AWS API Gateway

Die API Gateway API führt Requests zur Lambda Funktion.[3]

D. AWS CodeCommit

über CodeCommit können die Dateien für die statische Website dynamisch aktualisiert werden. Ausserdem nutzen wir ein CodeCommit Repository um den Code für die Lambda Funktion zu speichern.[4]

E. AWS CodePipeline

Code Pipelines sorgen dafür, dass Aktualisierungen in CodeCommit Repositories automatisch eingesetzt werden. [5]

REFERENCES

- [1] AWS S3
- [2] AWS Lambda
- [3] AWS API Gateway
- [4] AWS CodeCommit
- [5] AWS CodePipeline
- [6] Stock - Englisch für Aktie