

# 1 Einleitung

**TODO:** Überleitung zur Fragestellung überarbeiten.

In der heutigen Software-Entwicklung ist Open Source ein fester Bestandteil der Software Industrie. Von Frontend-Entwicklung über Datenbanken bis hin zu Machine Learning, überall kommen Open Source Bibliotheken, Frameworks und Software zum Einsatz. Im Frontend werden Frameworks bzw. Bibliotheken wie Angular oder React verwendet, bei der Auswahl von Datenbankmanagementsysteme stehen einem eine Vielzahl von Optionen zur Auswahl wie PostgreSQL, MySQL oder MongoDB und im Bereich Machine Learning werden Frameworks wie TensorFlow, Keras oder SciKit-Learn genutzt.

Doch worin unterscheiden sich diese und viele andere Erfolgreiche Open Source Projekte, von nicht so weit verbreiteten Wettbewerber. Mit dieser Bachelorarbeit soll die Frage beantwortet werden, *welche Faktoren Einfluss auf den Erfolg von Open Source Projekten, speziell in der JavaScript/TypeScript Umgebung haben.*

Mittels einer Datenerhebung von ausgewählten Open Source Projekten sowie einer Umfrage soll herausgefunden werden, welche Faktoren zum Erfolg eines Open Source Projektes beitragen. Ein zentraler Punkt dieser Ausarbeitung sind die extrinsischen sowie intrinsischen Anreize, die Nutzer zur Auswahl eines Produktes motivieren [Mid12]. Warum React und nicht Angular? Warum Debian und nicht Ubuntu etc.? Aspekte wie die interne Führung und Organisation der Projekte wird hierbei nicht thematisiert.

## 1.1 Definition von Open Source

Diese Arbeit folgt der gleichen Definition für Open Source wie von der *Open Source Initiative*<sup>1</sup> definiert wird. Entsprechend werden nur Projekte betrachtet die eine von der OSI genehmigten Lizenzen<sup>2</sup> nutzt.

## 1.2 Erfolg definieren

In der Literatur wird häufig in verschiedene Bereichen des Erfolgs unterteilt. Im Artikel von Midha und Palvia wird zwischen *Markterfolg* und *technischem Erfolg* unterschieden. Markterfolg definiert Midha et al. als Grad des Nutzerinteresses an ein Projekt, welches sich in der Beliebtheit des Projektes widerspiegelt. Den technischen Erfolg definiert Midha et al. durch die Entwickleraktivität, d.h. durch den Aufwand, den die Entwickler für das Projekt betreiben Beispielsweise die Häufigkeit und Frequenz von Updates und neuen Versionen. [Mid12].

In Steward et al. wird zwischen *Nutzerinteresse* und *Entwickleraktivität* unterschieden [Ste06]. Subramaniam et al. geht sogar weiter und unterteilt in *Nutzerinteresse*, *Entwicklerinteresse* und *Projekttätigkeit* [Sub09].

---

<sup>1</sup> <https://opensource.org/osd>

<sup>2</sup> <https://opensource.org/licenses/category>

## 1 Einleitung

Diese Bereiche werden getrennt betrachtet, da verschiedene Faktoren unterschiedlichen Einfluss auf den Erfolg eines Projektes haben. Während sich wachsendes Interesse bei Nutzern positiv auf den Markterfolg auswirkt, wirkt sich die Entwickleraktivität positiv den technischen Erfolg aus [Mid12, Ste06].

**TODO:** Footnote ausformulieren ggf link auf Glossar

**TODO:** Was sind GitHub Sterne?

In dieser Arbeit werden die Erfolgsfaktoren betrachtet, die zum Markt- bzw. technischen Erfolg eines Projektes beitragen. Der Markterfolg wird anhand Metriken gemessen wie: Downloads, Sterne auf GitHub sowie die Anzahl der Nutzer. Der technische Erfolg wird anhand von Metriken wie: Anzahl der Commits, Anzahl der Mitwirkenden am Projekt, Geschwindigkeit in der Tickets abgearbeitet werden und Verhältnis zwischen offenen/abgeschlossenen Issues<sup>3</sup> gemessen.

---

3 [Was sind Issues](#)

## 2 Erfolgsfaktoren

Es gibt viele Faktoren, die Einfluss auf den Erfolg eines Projektes haben. In diesem Kapitel werden die Faktoren in zwei Klassen aufgeteilt und näher erklärt. Die *Haupterfolgsfaktoren* werden in dieser Bachelorarbeit mittels Datenerhebung und Umfrage analysiert. *Weitere Faktoren* werden zusätzlich auf Basis von Literatur betrachtet und diskutiert. Zu den *Haupterfolgsfaktoren* gehören Eigenschaften wie *Lizenzen*, *Qualität*, *Dokumentation*, *Community*, *Sponsoren* und *Network Effekt*. Diese Faktoren alleine schaffen aber kein Gesamtbild des Erfolgs, deshalb wird im Kapitel 2.2 auf zusätzlich Aspekte eingegangen. Diese werden allerdings nicht mittels Datenerhebung oder Umfrage analysiert, da der Aufwand diese empirisch zu erfassen zu hoch wäre, stattdessen findet die Analyse auf rein literarischer Ebene statt. Hierzu gehören Faktoren wie *Das richtige Timing*, *Modularität und Komplexität*, *Responsibility Assignment*.

### 2.1 Haupterfolgsfaktoren

**TODO:** Einleitender Satz für Kapitel 2.1

#### 2.1.1 Lizenzen

Laut Subramaniam et al. spielen Lizenzen eine signifikante Rolle für den Erfolg von Open Source Software. Freie Lizenzen wie MIT oder BSD haben einen positiven Einfluss vor allem auf Software Entwickler. Denn Entwickler nutzen OSS, um es in eigene Projekte einzubauen, gegebenenfalls zu modifizieren und ein Endprodukt mit der OSS Komponente weiterzuverbreiten. Das ist mit restriktiven Lizenzen wie GPL meist nicht bedingungslos umsetzbar. Restriktive Lizenzen wie GPL wirken sich daher negativ bis neutral auf den Erfolg von OSS aus. Wenn die Software allerdings an Endnutzer gerichtet ist, wie zum Beispiel die Chat-App Telegram<sup>1</sup>, spielt die Lizenz eine weniger wichtige Rolle, da Weiterverbreitung und Modifizierung für diese Nutzergruppe keine Rolle spielen [Sub09].

Stewart et al. widerspricht der zweiten Aussage von Subramaniam et al. laut ihnen haben nicht-restriktive-Lizenzen nicht nur auf das Entwicklerinteresse, sondern auch auf das Nutzerinteresse einen positiven Einfluss. Während restriktive Lizenzen einen nicht signifikanten Einfluss auf Entwickleraktivität hätten. [Ste06]

Laut Midha und Palvia wirken sich freie Lizenzen positiv auf den Markterfolg aus, allerdings nur zu Beginn eines Projektes. Restriktive Lizenzen wiederum wirken sich negativ auf den technischen Erfolg aus [Mid12].

Ich glaube, dass sich offene Lizenzen durchgängig positiv auf ein Projekt auswirken. Offene Lizenzen werden tendenziell eher von Unternehmen verwendet als Projekte mit restriktiven Lizenzen, das führt zu einem dazu, dass die Beliebtheit und Bekanntheit des Projektes steigt, als auch die Wahrscheinlichkeit dass die Unternehmen zum Open Source Projekt etwas beitragen oder Sponsoren werden.

---

<sup>1</sup> <https://telegram.org/>

**H 1.** *Offene Lizenzen haben positiven Einfluss auf den Markterfolg.*

Steigt die Beliebtheit eines Projekts, so steigt auch das Interesse von Open Source Entwickler an einem renommierten Projekt mitzuwirken.

**H 2.** *Offene Lizenzen haben positiven Einfluss auf den technischen Erfolg.*

### 2.1.2 Gute Dokumentation

Dokumentationen spielen eine entscheidende Rolle für den Erfolg eines Projekts. Ohne eine gute Dokumentation ist die Software für die Benutzer als auch Contributor schwer zugänglich. Mailing Listen und StackOverflow können als eine gute Ergänzung zur Dokumentation dienen, allerdings kann diese dadurch nicht ersetzt werden [Ban13].

Eine gute Dokumentation ist auch ein Mittel, um neue Nutzer zu gewinnen. In Dagenais et al. heißt es, dass schon das Vorhandensein eines *Getting Started* Tutorials, den Nutzer beim Entscheidungsprozess positiv beeinflussen kann. [Dag10]. Ist man als Nutzer die ersten Schritte mit einer neuen Programmiersprache, Framework oder Bibliothek gegangen, steigt die Wahrscheinlichkeit, dieses Produkt auch zu nutzen. Beispielsweise hat die Web-Bibliothek ReactJS<sup>2</sup> auf der Homepage simple Beispiele, die man live editieren kann, ohne sich vorher etwas downloaden zu müssen, um einen ersten Eindruck von React zu gewinnen. Des Weiteren, gibt es ein sehr ausführliches Tutorial<sup>3</sup> welches über 5 Kapitel mit 21 Unterkapitel alle Grundbausteine von React abdeckt, um das gesamte Feature-Set in Kürze zu präsentieren und neue Entwickler von React zu überzeugen.

Laut einer GitHub Umfrage im Jahr 2017 sind unvollständige oder verwirrende Dokumentationen das größte Problem für Open Source Nutzer. Eine Gute Dokumentation hingegen lädt nicht nur neue Nutzer ein, sondern kann auch Nutzer zu Contributor machen. Sei es durch das Erstellen von Issues oder eines ersten Pull Requests. Hierfür spielen vor allem Contributing Guides und ein Code of Conduct eine wichtige Rolle, hierzu mehr im nächsten Kapitel [Git17].

Dokumentationen spielen eine wichtige Rolle, um neue Nutzer als auch neue Mitwirkende für ein Projekt zu gewinnen. Daher wird die Hypothese aufgestellt, dass Projekte mit guter Dokumentationen, erfolgreicher sind. Sowohl im Bereich des Markt- als auch technischen Erfolgs.

**H 3.** *Gute Dokumentationen ziehen mehr Nutzer an und führen so zu einem höheren Markt Erfolg.*

**H 4.** *Vorhanden sein von Contributing Guides und/oder Code of Conduct führt zu einem höheren technischen Erfolg.*

### 2.1.3 Eine Community Aufbauen

Ein weiterer Erfolgsfaktor in OSS ist die Community, sowohl die Community von Nutzern, als auch die Community von OSS-Entwickler. Es liegt in der Verantwortung der Projektleiter die Community aufzubauen und zu pflegen. [Ban13, Gita]. Auf GitHub finden sich Standardmäßig zwei Dateien, welche auf die Wahrnehmung dieser Verantwortung hindeutet. Die `CODE_OF_CONDUCT.md` und `CONTRIBUTING.md`.

---

<sup>2</sup> <https://reactjs.org/>

<sup>3</sup> <https://reactjs.org/docs/getting-started.html>

## 2 Erfolgsfaktoren

Das `CODE_OF_CONDUCT.md` ist ein Dokument, welches die Erwartungen an das Verhalten der Projektteilnehmer festlegt. Das Übernehmen und Durchsetzen des Code of Conducts kann dazu beitragen, eine positive und soziale Atmosphäre für alle zu schaffen [Gitc]. Häufig werden hierfür Vorlagen von der *Contributor Covenant Website*<sup>4</sup> verwendet.

**H 5.** *Das Vorhandensein eines Code of Conduct führt zu einem höheren Markterfolg.*

Die `CONTRIBUTING.md` Datei ist eine kurze Einführung, für potenzielle neue Contributor, wie man am jeweiligen Projekt mitwirken kann. Hier finden sich Anleitungen und Vorlagen für Bug Reports, Feature Requests, vorgehen bei Pull Requests, sowie Richtlinien bezüglich Coding Styles und Testabdeckung [Gitb]. Einige Projekte beginnen ihre `CONTRIBUTING.md` mit einem dank an den Leser und künftigen Contributor, wie beispielsweise Chakra-UI<sup>5</sup>, mit den Worten *"Thanks for showing interest to contribute to Chakra UI, you rock!"*. Somit wird die Hypothese aufgestellt, dass Projekte mit einem Contributing Guide eine höhere Chance für technischen Erfolg haben.

**H 6.** *Das Vorhandensein eines Contributing Guides führt zu einem höherem technischen Erfolg.*

### 2.1.4 Sponsoren **WIP**

### 2.1.5 Schneeball Effekt / Network Effekt **WIP**

Midha et al. fand eine starke Korrelation zwischen der vergangenen Beliebtheit eines Projekts, und der aktuellen Beliebtheit. Der Grund hierfür sei, dass die Beliebtheit als Entscheidungskriterium bei der Auswahl eines Projektes verwendet wird [Mid12].

Subramaniam et al. spricht von vom sogenannten *Network Effekt*, dieser wirkt sich laut [Sub09] positiv auf den Erfolg von OSS aus.

Das kann erfassen mittels Umfrage. Durch Fragen wie: "Wie wichtig sind Downloads und GitHub Sterne oder ähnliche Metriken bei der Auswahl eines Projektes, im Vergleich zu anderen Metriken."

**Hypothese in etwa:** Erfolgreiche Projekte werden noch erfolgreicher. => neue Projekte müssen raus stechen sonst haben sie keine Chance gegen die bestehenden OSS Alternativen

---

<sup>4</sup> <https://www.contributor-covenant.org/>

<sup>5</sup> <https://github.com/chakra-ui/chakra-ui>

## 2.2 Weitere Faktoren

**TODO:** Einleitenden Satz für Kapitel 2.2

### 2.2.1 Der richtige Zeitpunkt **WIP**

Direkt Zitat aus [Ban13] Kapitel 3.1

An interesting point made in Malcolm Gladwells book Outliers: The Story of Success [24] is that people are successful if their skills support products in a marketplace that is just maturing and where there is, consequently, still little competition. The same is certainly true for open source software projects as well: Projects that pick up a trend too late will have a difficult time thriving in a market that already supports other, large and mature projects

### 2.2.2 Modularität und Komplexität **WIP**

Bezüglich Modularität gibt es hier etwas mehr [Mar15] siehe Kapitel IV. A) beziehungsweise die Quellen [42,43,49] in [Mar15]

# Literaturverzeichnis

- [Ban13] W. Bangerth und T. Heister. What Makes Computational Open Source Software Libraries Successful? *Computational Science & Discovery*, 6(1):015010, Nov. 2013.
- [Dag10] B. Dagenais und M. P. Robillard. Creating and Evolving Developer Documentation: Understanding the Decisions of Open Source Contributors. In *Proceedings of the Eighteenth ACM SIGSOFT International Symposium on Foundations of Software Engineering, FSE '10*, S. 127–136. Association for Computing Machinery, New York, NY, USA, Nov. 2010.
- [Gita] Building Welcoming Communities. <https://opensource.guide/building-community/>. Zuletzt aufgerufen am 02.04.2022.
- [Gitb] Starting an Open Source Project. <https://opensource.guide/starting-a-project/>. Zuletzt aufgerufen am 07.04.2022.
- [Gite] Your Code of Conduct. <https://opensource.guide/code-of-conduct/>. Zuletzt aufgerufen am 07.04.2022.
- [Git17] Open Source Survey. <https://opensourcesurvey.org/2017/>, 2017. Zuletzt aufgerufen am 03.04.2022.
- [Mar15] D. Margan und S. Čandrlić. The success of open source software: A review. In *2015 38th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, S. 1463–1468. 2015.
- [Mid12] V. Midha und P. Palvia. Factors Affecting the Success of Open Source Software. *Journal of Systems and Software*, 85(4):895–905, Apr. 2012.
- [Ste06] K. J. Stewart, A. P. Ammeter und L. M. Maruping. Impacts of License Choice and Organizational Sponsorship on User Interest and Development Activity in Open Source Software Projects. *Information Systems Research*, 17(2):126–144, 2006.
- [Sub09] C. Subramaniam, R. Sen und M. L. Nelson. Determinants of Open Source Software Project Success: A Longitudinal Study. *Decision Support Systems*, 46(2):576–585, Jan. 2009.