

# 1 Umfrage

Als Ergänzung zur statischen Datenerfassung der GitHub Projekte wurde auch eine Umfrage durchgeführt. Ziel der Umfrage war es herauszufinden worauf die Nutzer von Open Source achten, wenn sie ein Projekt wählen. Das übergeordnete Ziel war es die Hypothesen H3, H6 und H8 zu prüfen. An der Online-Umfrage haben Software Entwickler der adesso, sowie Studenten der TH Rosenheim teilgenommen, insgesamt erhielt die Umfrage 308 Antworten.

Hierbei sollten die Teilnehmer Aussagen zu den Theme Dokumentation, Beliebtheit, Sponsoren und Entwicklung bewerten. Als Bewertungsskala wurde eine 5-Punkte Likert Skala verwendet [Lik32].

## 1.1 Dokumentation

Im ersten Teil der Umfrage mussten die Teilnehmenden zwei Aussagen bewerten.

Die erste Aussage *"Was macht gute Dokumentation aus?"*, mit den **Punkten**:

- Übersichtlichkeit
- Einfache Sprache
- Live-Demos
- Übersetzungen (z.B. Englisch -> Deutsch)
- Das Vorhandsein einer Getting Started Page
- Code Beispiele
- Strukturierung der Dokumentation

Die zweite Aussage war *"Was macht schlechte Dokumentation aus?"*, mit den **Punkten**:

- Keine Code Beispiele
- Schlecht strukturierte Dokumentation
- Dokumentation zu komplex
- Fehlende Übersetzung (z.B. fehlende deutsche Übersetzung)
- Fehlende Getting Started Page

Diese Punkte wurden gewählt basierend auf Beobachtungen vieler Dokumentationen, während des "Einsammel"prozesses sowie Allgemein erfahrung die ich als Entwickler hab. Manche Dokumentationen sind strukturierter als andere, Manche haben Code-Beispiele manche nicht, manche haben Live-Demos manche nicht. Hiermit wollte ich quasi herausfinden welche punkte die wichtigsten sind für die Befragten. Wie sich herausstellt sind es Code-Beispiele.....

**TODO:** Etwas genauer erklären was hier gewichtet heißt

Die Aussagen wurden auf einer 5-Punkte Skala bewertet, hierbei entspricht 1 *Stimme gar nicht zu* und 5 *Stimme vollkommend zu*. In den Tabellen ?? und ?? finden sich die Antworten auf die jeweiligen Aussagen als gewichteten Mittelwerte.

Die Umfrage hat gezeigt, dass Code Beispiele und Übersichtlichkeit bzw. Struktur die wichtigsten Eigenschaften einer guten Dokumentation sind. Die Teilnehmer bewerteten diese als wichtige Aspekte einer guten Dokumentation, während die Abwesenheit dieser Aspekte tendenziell Nutzer motivieren würde sich nach Alternativen umzuschauen. Eine Getting-Started Page wird tendenziell als wichtig

## 1 Umfrage

empfunden, dessen Abwesenheit wird allerdings weniger kritisch betrachtet als die vorher genannten. Eine Getting-Started Page bewegt sich basierend auf diesen Daten daher eher in die Richtung *Nice-To-Have*, während Code Beispiele und Struktur *Must-Have* sind.

Zudem gab es jeweils ein Freitextfeld, wo die Teilnehmenden die Möglichkeit hatten weitere Aspekte einer guten bzw. schlechten Dokumentation zu nennen. Diese Freitextfelder waren optional und wurden von 95 bzw. 71 von den gesamten 308 Teilnehmenden genutzt.

In den Tabellen 1.3 (a) und (b) finden sich die am häufigsten genannten Punkte der Freitextfelder. Aktualität, Vollständigkeit und UX waren die am häufigsten genannten Punkte bezüglich guter Dokumentation, die jeweiligen pendants, veraltete Dokumentation, unvollständige Dokumentation und schlechte UX waren die häufigsten genannten Gründe, um ein OSS Projekt nicht zu nutzen. Somit ergänzen diese drei Punkte die vorhin erwähnten *Must-Haves*.

**TODO:** Diesen Fact etwas schöner einbinden

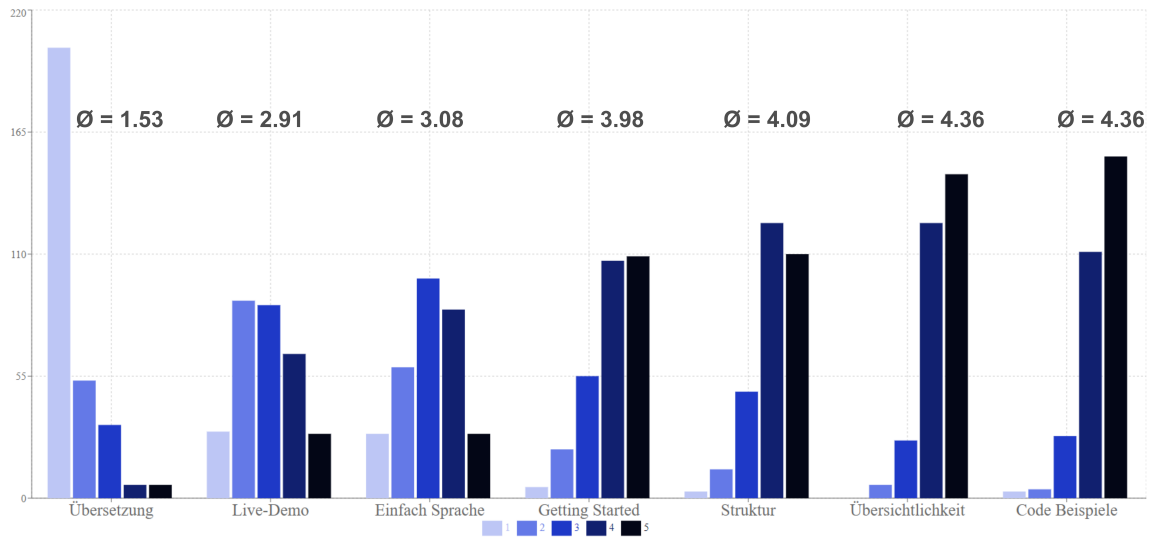
Die letzte Frage zum Thema Dokumentation war "*Hat eine schlechte Dokumentation Sie jemals dazu gebracht, ein alternatives Projekt zu wählen?*".

81% der Teilnehmer haben diese Frage mit "Ja" beantwortet.

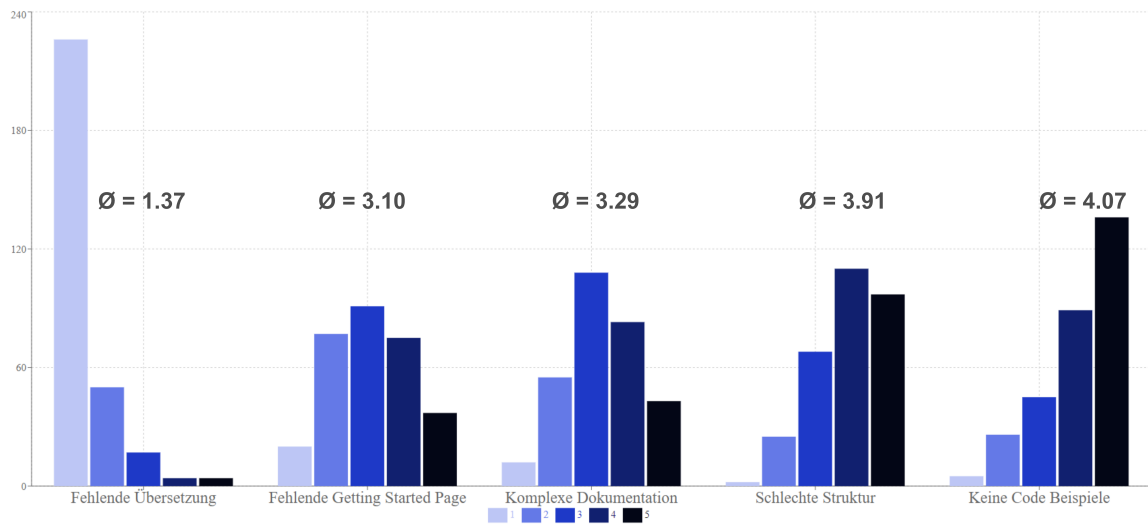
### Anmerkung:

Im Fall von UX wurden folgende Punkte zusammengefasst: Suchfunktion/Verlinkung, Design und Übersichtlichkeit.

## 1 Umfrage



**Abbildung 1.1** Antworten: Was zeichnet Gute Dokumentation aus?



**Abbildung 1.2** Antworten: Was zeichnet Gute Dokumentation aus?

## 1 Umfrage

**TODO:** Die zwei Tabellen unten werden als **Abbildung** gekennzeichnet ! Statt Tabellen Copy Paste einfach die Tabelle 4

Aktualität	28
Vollständigkeit	21
Gute UX	14
Versionierung	10
Changelog vorhanden	6
Gute Code Beispiele	5

**(a)** Freitext: Gute Dokumentation

Veraltet	25
Unvollständig	15
Schlechte UX	11
Fehlerhaft	7
Keine Doku vorhanden	6
Code Beispiele Funktionieren nicht	5

**(b)** Freitext: Schlechte Dokumentation

**Abbildung 1.3** Häufigste Erwähnungen beim Freitext

## 1.2 Letzte Frage

**TODO:** Formulierung

In der letzten Frage wurden die Teilnehmer aufgefordert Kriterien bei der Auswahl von OSS zu nach Wichtigkeit sortieren.

## 1.3 Erkenntnisgewinn der Umfrage

**TODO:** Hier auf Hypothesen eingehen etc...

## 2 Diskussion

### 2.1 Zu Hypothese 1

Mit dieser Datenerhebung hat sich Hypothese H1 bestätigt. Allerdings muss man einräumen, dass für die Beliebtheit nur GitHub Sterne verwendet wurden. Der Linux Kernel ist GPLv2 lizenziert und gehört zu den bekanntesten und erfolgreichsten Open Source Projekten. Ein Grund weshalb freizügige Lizenzen beliebter sein könnten ist, dass es sich bei restriktiven Projekten meist, um Applikationen handelt. Telegram, Signal und ähnliche Open Source Chat Apps werden von Entwicklern sowie nicht Entwicklern verwendet, die eventuell gar keine GitHub Accounts haben. GitHub Sterne erfassen somit nicht das gesamte Bild *Markterfolg*.

### 2.2 Zu Hypothese 3

Für manche Projekte lohnt es sich ggf. nicht ein Code of Conduct zu haben, da die Teams schlichtweg zu klein sind. Der Code of Conduct muss nicht nur aufgestellt sondern auch umgesetzt werden. Es kann also sein, dass einige Projekte den CoC erst einführen wenn es sich lohnt, sprich ab einer gewissen Größe erst... Wieder nur ein Vergleichsmaß von Sternen (aber auch Downloads so thats something)

### 2.3 Zur Hypothese mit Sponsoren...

#### Anmerkung

Man muss allerdings anmerken, dass einige Projekte von Unternehmen entwickelt werden. Diese finanzieren ihre Projekte meist selbst und sind somit nicht abhängig von Sponsoren. React beispielsweise kommt aus dem Hause Facebook, während VueJS unabhängig ist und auf Sponsoren angewiesen ist. In diesem Kontext zählen Projekte von Unternehmen auch als gesponsert. Es sind aber auch nur 10 Projekte von Unternehmen, also fällt das auch nicht so sehr ins Gewicht tatsächlich.

- Stichprobe war klein
- (Problem => Donations not enough, abhängig was aus dem Crawler für Ergebnisse kommen)
- Evtl ist die Umfrage Stichprobe sehr spezifisch (Überwiegend Studenten)
- Fazit => Problem mit Donations? Zu wenige? Findet man dieses Problem in den Daten überhaupt... existiert dieses Problem überhaupt Soll allerdings nur angeschnitten werden, da die BA nur auf die Sicht der Nutzer geschaut hat. Probleme der Open Source Entwickler ist ein anderes Problem.
- Die Daten aus dem *Crawler* sind nur ein Snapshot der aktuellen Lage, sprich man kann nur sagen. Ja es gibt eine Korrelation zwischen z.B. Sponsoren und User Interest, aber nicht was was hervorruft. Hat das Projekt Sponsoren weil es viele Nutzer hat *oder* hat das Projekt viele Nutzer weil es Sponsoren hat? Dafür bräuchte man eine Datenerhebung over time. Diese ist allerdings im kleinen Zeitfenster einer Bachelorarbeit nicht machbar. [Mid12] hat beispielsweise in einem Zeitfenster von 3 Jahren die Datenerhebung gemacht.

### 2.3.1 Disclaimer

<https://www.npmjs.com/package/npm-downloads-increaser>

npm downloadzahlen sind manipulierbar, ich geh zwar davon aus, dass es allgemein nicht häufig gemacht wird und da ja sowieso vielmehr in die rechnung rein geht als nur npm downloads. Werden diese daten trotzdem mit erfasst. [How to exploit NPM](#).

In einem [Blogpost](#) erklärt NPM wie es zu den Download zahlen zustande kommt.

# Literaturverzeichnis

- [Lik32] R. Likert. A technique for the measurement of attitudes. *Archives of psychology*, 1932.
- [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.