

Zusammenarbeit

Inhalt

1. Teilnahme
2. Mitarbeit
3. Kommunikationskanäle
4. Tools
5. Freiwillige Arbeit vs. bezahlte Arbeit
6. Take Home Message
7. Quellen

1. Teilnahme

Viele Personen, die sich dazu entschlossen haben an einem Open-Source-Projekt teilzunehmen, wissen oftmals zu Beginn nicht, an welchem Projekt sie teilnehmen möchten. Am geeignetsten sind Projekte, die der Entwickler als Nutzer (engl. User) schon kennengelernt hat, ansonsten kann man auch im Internet nach anderen Projekten Ausschau halten. Gute Referenzen sind *FLOSS Weekly*, *GitHub*, *Sourceforge*, etc.

Am Anfang ist es wichtig die Community kennenzulernen. Man sollte herausfinden, wie die Community kommuniziert bzw. welche Kommunikationskanäle genutzt werden, und wie sie geregelt ist (Mailingliste eintragen, dem Blog folgen, IRC Channel beitreten). Generell sollte man sich zu Beginn einen Überblick über das Projekt verschaffen. Am besten man teilt zugleich der Community mit, welche Absichten man hegt, und welche Fähigkeiten man besitzt (Lester, 2014; Yeates, 2005).

2. Mitarbeit

Um an einem Open-Source Projekt teilnehmen zu können ist es nicht zwingend nötig ein guter Programmierer zu sein. Natürlich werden sie auch gebraucht aber man kann schon vieles bewirken wenn man nicht gut oder gar nicht programmieren kann (vgl. Tabelle unten).

<p>mit Tickets arbeiten:</p> <ul style="list-style-type: none"> ➤ einen Bug diagnostizieren, ➤ fixed Bugs schliessen 	<p>mit Dokumentation arbeiten:</p> <ul style="list-style-type: none"> ➤ ein Beispiel schreiben, ➤ Wiki, ➤ Developer Guide,
<p>mit Code arbeiten:</p> <ul style="list-style-type: none"> ➤ Bug report ➤ neue Features designen, bestehende Features redesignen bzw. weiterentwickeln ➤ Eine Beta Version testen, ➤ einen Bug fixen, ➤ einen Test schreiben, ➤ Compiler Warnungen ausblenden (C basierte Projekte), ➤ Code kommentieren 	<p>mit der Community arbeiten:</p> <ul style="list-style-type: none"> ➤ Übersetzungen, ➤ Grafikarbeiten → icons, logos, banners, etc., ➤ User Support → Fragen beantworten, FAQs, HOW TOs, Guides, Tutorials schreiben, ➤ Blog Post schreiben, ➤ die Website verbessern, ➤ Bei der Verbreitung des Projekts helfen, ➤ Feature Request erstellen

Table 1: Übersicht über verschiedene Bereiche in einem Open-Source-Projekt, an denen man mitarbeiten kann (Lester, 2012; Tatham, 2010)

3. Kommunikationskanäle

Die wichtigsten Kommunikationskanäle in einem Open-Source-Softwareprojekt stellen **Website**, **Mailinglisten**, und **IRC Channels** dar. Die Website dient v.a. in einem ersten Schritt als Einstieg in das Projekt. Von dort aus ist idealerweise alles zugänglich. Man findet dort auch Informationen darüber, was der momentane Stand des Projektes ist, und was für nächste Schritte geplant sind. In der Mailingliste werden meist aktuelle Ideen, Designvorschläge für das Projekt diskutiert. In grösseren Projekten werden sind auch verschiedene Mailinglisten zu verschiedenen Unterthemen verfügbar. IRC Channels bieten ergänzend Echtzeitdiskussion an, und haben noch weitere Vorteile wie, schnellere Bindung zu der Community, Schnelligkeit, Echtzeitmeetings.

Weitere Sekundäre Kommunikationsmittel, die in Open-Source-Projekten genutzt werden sind Foren (v.a. für Support), Blogs, RSS Feeds und soziale Netzwerke (Bacon, 2009; Fogel, 2005; Gardler, 2011).

4. Tools

Version Control Systeme (VCS) und *Issue Tracking Systeme* sind Tools, die die Zusammenarbeit in einem Open-Source Projekt garantieren. Eine VCS dient dazu Änderungen am Quellcode zu erfassen. Alle Versionen werden mit Zeitstempel und Benutzerkennung archiviert und gesichert. Somit können ältere Versionen wiederhergestellt werden (Wikipedia, n.d.-b). Issue Tracking Systeme dienen als Werkzeug, um neben Programmierfehler auch Verbesserungsvorschläge und Wünsche der Nutzer zu erfassen und zu dokumentieren (Wikipedia, n.d.-a).

5. freiwillige Arbeit vs. bezahlte Arbeit

„Every good work starts by scratching a developer’s personal itch, (Raymond, 2008).“

Die Bewegungsgründe von Freiwilligen, die an einem Projekt teilnehmen, und den bezahlten Softwareentwicklern, die von Firmen eingesetzt werden, unterscheiden sich qualitativ (vgl. Tabelle unten).

Freiwillige:	bezahlte Softwareentwickler/ Firmen:
<ul style="list-style-type: none"> ➤ Eigene Bedürfnisse/ Wünsche in die Tat umsetzen ➤ Das Gefühl von Bedeutung ➤ Teilen ➤ Helfen ➤ Bewusstsein für die Bedürfnisse des Users erlangen ➤ Das Gefühl haben, dabei zu sein 	<ul style="list-style-type: none"> ➤ Verteilung der Arbeitslast ➤ Einen Service verbessern ➤ Hardware Verkäufe erhöhen ➤ Konkurrenz untergraben

Tabelle 2: Beweggründe von Freiwilligen und bezahlten Softwareentwicklern im Vergleich (Fogel, 2005; Hardy, 2013)

6. THM

- ✓ Jeder kann etwas zu einem Projekt beitragen
- ✓ Website, Mailinglist und IRC-Channel als wichtigstes Kommunikationsmittel
- ✓ Issue-Tracking und VCS sind wichtige Tools, die die Zusammenarbeit garantieren

7. Quellen

Bacon, J. (2009). *The art of community*. O’Reilly Media.

Fogel, K. (2005). *Producing Open Source Software How to Run a Successful Free Software Project*.

Gardler, R. (2011). Essential tools for running a community-led project. Retrieved September 24, 2014, from <http://oss-watch.ac.uk/resources/communitytools>

Hardy, V. (2013). Why Do Developers Contribute to Open Source Projects? Retrieved September 24, 2014, from <http://www.websitemagazine.com/content/blogs/posts/archive/2013/08/29/why-developers-contribute-to-open-source-projects.aspx>

Lester, A. (2012). 14 Ways to Contribute to Open Source without Being a Programming Genius or a Rock Star. Retrieved September 24, 2014, from <http://blog.smartbear.com/programming/14-ways-to-contribute-to-open-source-without-being-a-programming-genius-or-a-rock-star/>

Lester, A. (2014). The Beginner’s Guide to Contributing to Open Source Projects. Retrieved September 26, 2014, from http://blog.newrelic.com/2014/05/05/open-source_gettingstarted/

Raymond, E. S. (2008). *The Cathedral & the Bazaar: Musings on Linux and Open Source by an Accidental Revolutionary*. O’Reilly Media; Auflage: 1.

Tatham, E. (2010). Roles in open source projects. Retrieved September 25, 2014, from <http://oss-watch.ac.uk/resources/rolesinopensource>

Wikipedia. (n.d.-a). Bugtracker. Retrieved September 29, 2014, from <http://de.wikipedia.org/wiki/Bugtracker>

Wikipedia. (n.d.-b). Versionsverwaltung. Retrieved September 29, 2014, from <http://de.wikipedia.org/wiki/Versionsverwaltung>

Yeates, S. (2005). A guide to participating in an open source software community. Retrieved September 25, 2014, from <http://oss-watch.ac.uk/resources/toptipscommunities>