

Vorlage für eine Seminararbeit

NACHNAME, VORNAME
Hochschule Mannheim
Fakultät für Informatik
Paul-Wittsack-Str. 10, 68163 Mannheim

Abstract—Outsourcing work is a well known by companies. Software crowdsourcing has a lot of potential to become a major technology in software development. It provides motivations as well as barriers to newcomers and companies. This paper is a systematic literature review. It explains different methods and types of software crowdsourcing. Also it reveals barriers faced by newcomers as well as it show the motivation, from a contributor and from a company view, to take part in an software crowdsourcing / open source software project / FLOSS... The study provides insights for tool designers, managers and researchers, who want to improve the overall amount and quality of contributing in their projects.

Contents

1	Introduction	1
2	Research Method	1
3	State of the art	1
3.1	Crowdsourcing models	2
3.2	Levels of Crowdsourcing	2
4	Motivation for software crowdsourcing	2
4.1	Stakeholder Perspectives	3
4.2	Contributor Perspectives	3
5	The process of joining a OSS project	3
6	barriers faced by newcomers	4
7	Support newcomer to overcome the barriers's	5
8	conclusion and future work	5

1. Introduction

Software Crowdsourcing had a huge popularity rise in the last few years, even it exists for a very long time. For community projects, such as Linux or Firefox, it is a well known and existing dependant technique. Without the crowd such projects, e.g. elementary OS, Apache and many more would not be possible. With the rise of software crowdsourcing many people try out crowdsourcing. Before somebody can contribute to an project there are several barriers, which specially newcomers has to overcome. Similiar to software crowdsourcing there are other techniques to get results from the crowd. Those

technique are; Free Libre Open Source Software (FLOSS), Free Open Source Software (FOSS) and Open Source Software (OSS). Due to those techniques are all very similiar to software crowdsourcing, all of them are included in the search for barriers newcomers are facing before they successful contribute in such an project.

2. Research Method

To find relevant scientific papers i used search engines, 'GoogleScholar', 'IEEE Xplore', 'ACM Digital Library' and 'Springer Link'. Moreover, I focused mainly on re-searches that are not older than 2016(really?!?), because there were not many papers focusing on the barriers newcomer has to overcome I also selected some older reasearches. Many reasearches I found did focus on other topics, such as In the end I was able to find papers, where I could use of them. Also I found some literature research papers, which I used to find new papers as well as to match the results of the literature research.

3. State of the art

- Patterns for tearing down contribution barriers to FLOSS projects[24]
onion model
- Cloud-Based Software Crowdsourcing[23]
- Crowdsourcing in Software Engineering: Models, Motivations, and Challenges[7]
- The Economic Motivation of Open Source Software: Stakeholder Perspectives[12]
community vs commercial
- Understanding the Impressions, Motivations, and Barriers of One Time Code Contributors to FLOSS Projects: A Survey[9]
Successful FLOSS projects must attract and retain these peripheral developers so they can provide more significant code contributions.[9, 14]
- Understanding and Supporting the Choice of an Appropriate Task to Start with in Open Source Software Communities [14]

Nowadays crowdsourcing can be found in a lot of different areas as the system gets more an more popular. Due to the popularity of crowdsourcing it is no incident that you will find a variety of crowdsourcing platforms, e.g. Idea Bounty, CrowdSpring, 99Designs, TopCoder and many more. Due to the high use of the word it is not possible to coherently classify the term 'crowdsourcing'[1]. As

Enrique Estellés-Arolas and Fernando González-Ladrón-de-Guevara in the article 'Towards an integrated crowdsourcing definition'[1] wrote:

The name crowdsourcing is formed from two words: crowd, making reference to the people who participate in the initiatives; and sourcing, which refers to a number of procurement practices aimed at finding, evaluating and engaging suppliers of goods and services.

3.1. Crowdsourcing models

During the literature review I found four different types of crowdsourcing models. Within those four models there are two major ways of software crowdsourcing. The competitive method, which had a incredible increase during the last years. The oldest method is the collaborative, which has its roots in the software crowdsourcing and is still a huge factor if or if not crowdsourced projects are successful.

The first model is the competitive, also called commercial, model. In this model the collaborator are contestants to each other[7, 12]. A natural motivation in this model is to compete against each other and the winner will get a reward of the company. With those two motivations it is no wonder that this model gained a lot of attention in software development. The most popular platform for this model is TopCoder (www.topcoder.com) with over one million people in the community and over seven thousand challenges per year. An old model is the 'Peer Production', it is also called to be the community version of SW CS [7, 12]. 'Peer Production' is an open source procedure which consists of software projects where the community contributes to their favorite project. There is a huge amount of these community open source projects nowadays. Some of the most known are Linux, Firefox, Apache and Atom. Those projects are great example how the crowd can create a great product. Also this model depends that a lot of people contribute to the project. Like the onion model of free libre open source software (FLOSS) projects the onion model for this crowdsourcing would look similar. At the core there would be the innovation leaders, those who lead the project. Around them there would be the key contributors, those contributors who work constantly on the project. Those contributors who are just active sometimes, e.g. to fix bugs, are at the outside of the onion model. Another model is called 'Microtasking'[7]. In this model the tasks will split up to several small task. With this procedure there will be a lot of small tasks. Without much time needed to complete one task, many contributors are up to do the task. With this redundancy there will be also some differences in the solutions. Due to those solutions the requester can select the for him best suited solution. A similar to the previous model is the 'Economic model'[23]. The goal of this model is also to maximize the contributions. The companies are creating strategies to include the crowd. Also, the winner, sometimes also the best three solutions, will receive a reward for his contribution.

3.2. Levels of Crowdsourcing

The ideas of this subsection were received by the article 'Cloud-based Software Crowdsourcing (Dagstuhl Seminar 13362)'[6] written by Huns et al. Due to it was the only paper which described the different levels of crowdsourcing, the ideas will be from this article.

As well there are different method of crowdsourcing there are also different level's. Those levels describe how many contributors work on the contribution as well as how large the task is and how much time the contributors have. Those level's are called 'Level 1' up to 'Level 4', where the first level is describes the individual developer. In this level the committers have clear tasks, which they have to solve within a few months. Also, the individuals developers as well as the projects can be ranked. The second level uses small teams with up to ten people per team. Those teams have up to one year time, depending on the project, to develop their solution for the given task. Also it allows that the developers will get feedback and thoughts from the stakeholders and the community during the developing process. The next larger level has teams of more than 10 and up to 100 people and a development time of up to two years. Due to this large time span and the larger teams the crowdsourcing platforms for 'Level 3' crowdsourcing need to have an automated cross-verification and is not suited for commercial crowdsourcing, as well as the largest level. With this technique the platform will automatically check if the requirements are met or not. The largest level has international factors. The developers, from several countries, develop a large and flexible system. Due to the international factors, those projects has to overcome additional barriers before they can succeed. Those barriers are social-cultural barriers, which will be explained in Section X.

4. Motivation for software crowdsourcing

As humans we have two types of motivation. One is the 'intrinsic motivation' and the other one is the 'extrinsic motivation'. Intrinsic Motivation describes the motivation of doing a task for inborn satisfaction[13]. Regarding to software crowdsourcing this could be to contribute with reasons like; fun and enjoyment, technical learning, altruism. The opponent of the intrinsic motivation is the extrinsic motivation. Extrinsic motivation play always a role, if the person gets something afterwards to do the task [13]. Examples therefor are the career status, reputation, fame, community identification. All of the motivations showed in this section can be defined as intrinsic or extrinsic motivation. Also, in the table 1 you will see that there a slightly more intrinsic motivations than extrinsic motivations from a contributors view. As the subsection "Contributer Perspectives" shows, the amount of both motivation types is similar. This similarity is important, because to much of one motivation type can have negative effects on the other type [5]. From a stakeholder views there are only extrinsic goals, as shown in table 4.1. For companies the similarity of both motivations does not count, because the main goal of a company is to get something in return for their work.

4.1. Stakeholder Perspectives

The perspectives for the stakeholders were found in the following papers;

- Riehle; The Economic Motivation of Open Source Software: Stakeholder Perspectives[12]
- LaToza and van der Hoek; Crowdsourcing in Software Engineering: Models, Motivations, and Challenges[7]
- Hossain; Crowdsourcing: Activities, incentives and users' motivations to participate[4]
- Tsai et al.; Cloud-Based Software Crowdsourcing[23]

As a stakeholder views, there are in depth reasons why you should consider investing in the open source market. The companies, which are using the open source market, do not sell the product. More likely, they sell the provision, maintenance and the support for the product. Another motivation is; the companies consider more the opinions of the crowd and let them choose the task to work so. By doing so, more people of the crowd will contribute and also they will find more qualitative solutions to their task. Due to the last two points, the customers will trust your company and the product more. With this in the knowledge the company will have less marketing costs. Also, the number of customers will increase rapidly. The companies can hire and fire employees more easily, because there is an large pool of people who fit to the company or did already worked on the project. Moreover, it is easier to find and employ specialists, due to every project has its 'core worker'. Those core worker are worker, who are specialists in the projects and also work a long time on it. So the core worker have a lot of knowledge and it is also shown that they lead innovations in the projects[XY, 4]. As a result of finding an employing specialists those specialists can develop more innovative products. With the diversity of the crowd, companies will get alternative solutions. When a company start to enter the open source software field, they will have less time until the next innovative idea comes up. Also, they will have variable costs. It will be easier to share information and they will find an increase in the development speed. The increase in the development speed is found due to the crowd, as they overtake parts of the development. By spending a longer time on the open source market, your customers will become more loyal and the companies will find less risks by launching a new product. Also, the companies will have an increase of the quality of their software. This is a result of the trust by the customers that the company provides, maintain and support their product for a longer time. All those motivations are listed in the table 4.1.

4.2. Contributor Perspectives

The motivations for this subsection were found in the following papers;

- Hannebauer et al.; An Exploratory Study of Contribution Barriers Experienced by Newcomers to Open Source Software Projects[2]
- Hannebauer and Gruhn; On the Relationship Between Newcomer Motivations and Contribution Barriers in Open Source Projects[3]

- Lee et al.; Understanding the Impressions, Motivations, and Barriers of One Time Code Contributors to FLOSS Projects: A Survey[9]
- LaToza and van der Hoek; Crowdsourcing in Software Engineering: Models, Motivations, and Challenges[7]
- Hossain; Users' motivation to participate in online crowdsourcing platforms[5]

A more detailed view and who wrote about which paper will be shown in figure XY.

The motivations for the contributors were listed in several papers. The most named motivation is; improvement of programming skills or learn a new technology. I combined those two motivations because the goal of both is to improve yourself, no matter it is by increasing the developing skills or by learning a complete new developing language. Close to this motivation is that contributors want to learn about the project technologies. The second most motivation I found, is the reason that the contributors need a modification for their own project. Similiar to this motivation I found also that a motivation is to make the project interoperable with one's own products. As often as the motivation before, contributors mentioned also that they enjoy programming. Similar to the contributors enjoy programming there are also contributors who enjoy the intellecual challenge. Some contributors mentioned also that they want to give something back to the community. Close to the last motivation, to consult FLOSS projects was also found. The belief in the idea of open source projects were also named as motivation. Also evaluating a project was found as a motivation to contribute and to give feedback to the project community. Try out open source software projects and to join a community was also mentioned as motivation. By submitting patches, satisfaction will derive from working in a project and with the community. Also, that the changes will be reviewed by others is also a motivation to contribute. A missing feature is also called to be a motivation. To gain the respect of the community was found bei Hannebauer. Personal visibility and similiar to it reputation were also revelead to motivate contributors. Some contributors maybe get paid to contribute, others may get a reward by submitting. The last motivation I could find is that the contributors will have only to contribute once. They will not have to rebuild their patch for the next version of the project again. As well as the stakeholder motivations, those motivations will be divded into intrinsic and extrinsic motivation and listet in table 1. With this table you will get a good overview of the contributor motivations.

5. The process of joining a OSS project

- Newcomers lurk on mailing lists, read previous discussions, and review public information to learn about people, ongoing activities, history and current status of the community. Lurking appears to be an essential part of the joining process, as newcomers acquire the social and technical knowledge expected of participants [31, 36, 42, 43]. [Beyond Pretty Pictures Examining The Benefits of Code Visu...]
- After acquiring some knowledge about the project and its norms, newcomers start interacting with

Motivation	Motivation type	Found X times	Papers
Extrinsic Motivation			
the company will have less marketing costs	Extrinsic	2	[4, 23]
it is easier to find and employ specialists	Extrinsic	2	[7, 23]
specialists can develop more innovative products	Extrinsic	2	[4, 23]
companies will get alternative solutions	Extrinsic	2	[7, 23]
the company sell the provision, maintenance and the support for the product	Extrinsic	1	[12]
the company find more qualitative solutions to their task	Extrinsic	1	[7]
the number of customers will increase rapidly	Extrinsic	1	[4]
the companies can hire and fire employees more easily	Extrinsic	1	[12]
the company will have less time until the next innovative idea comes up	Extrinsic	1	[4]
the company will have variable costs	Extrinsic	1	[7]
the customers will become more loyal	Extrinsic	1	[4]
the companies will find less risks by launching a new product	Extrinsic	1	[4]
the companies will have a increase of the quality of their software	Extrinsic	X	XY

Table 1. MOTIVATIONS TO CONTRIBUTE TO A SOFTWARE CROWDSOURCING PROJECT

Motivation	Motivation type	Found X times	Papers
Intrinsic Motivation			
improvement of programming skills or learn a new technology	Intrinsic	4	[2, 3, 9, 7]
enjoy programming	Intrinsic	3	[3, 5, 7]
want to give something back to the community	Intrinsic	3	[2, 5, 7]
want to learn about project technologies	Intrinsic	1	[2]
enjoy the intellectual challenge	Intrinsic	1	[2]
to consult FLOSS projects	Intrinsic	1	[2]
The belief in the idea of open source projects	Intrinsic	1	[3]
evaluating a project	Intrinsic	1	[2]
try out open source software projects	Intrinsic	1	[3]
to join a community	Intrinsic	1	[3]
satisfaction will derive from working in a project and with the community	Intrinsic	1	[2]
the contributors will have only to contribute once	Intrinsic	1	[xy]
Extrinsic Motivation			
need a modification for their own project	Extrinsic	3	[2, 3, 9]
personal visibility and reputation	Extrinsic	2	[2, 5]
make the project interoperable with one's own products	Extrinsic	1	[2]
the changes will be reviewed by others	Extrinsic	1	[2]
gain respect of the community	Extrinsic	1	[2]
get a reward or paid to contribute	Extrinsic	1	[9]

other members by joining discussions and take part in peripheral activities such as reporting or fixing bugs. [Beyond Pretty Pictures Examining The Benefits of Code Visu...]

- Observations of newcomers show that successful joiners do not typically start by introducing themselves or expressing their interest in contributing. Instead, they participate in ongoing technical discussions about existing problems [14, 42]. [Beyond Pretty Pictures Examining The Benefits of Code Visu...]
- Their initial messages are not limited to suggesting technical solutions for problems but often include code, suggesting a high degree of technical knowledge and skills is required [Beyond Pretty Pictures

Examining The Benefits of Code Visu...]

6. barriers faced by newcomers

- Understanding the Impressions, Motivations, and Barriers of One Time Code Contributors to FLOSS Projects: A Survey[9]
- Understanding and Supporting the Choice of an Appropriate Task to Start with in Open Source Software Communities[14]
- An Exploratory Study of Contribution Barriers Experienced by Newcomers to Open Source Software Projects[2]
- Barriers Faced by Newcomers to Software-Crowdsourcing Projects[25]

- On the Relationship Between Newcomer Motivations and Contribution Barriers in Open Source Projects[3]
- The Good, the Bad and the Ugly: An Onboard Journey in Software Crowdsourcing Competitive Model[10]
- The Hard Life of Open Source Software Project Newcomers[19]
- Why do newcomers abandon open source software projects?[22]
- Patterns for tearing down contribution barriers to FLOSS projects[24]
- Labeling relevant skills in tasks: Can the crowd help?[8]
- SLR's

A systematic literature review on the barriers faced by newcomers to open source software projects[16]

Barriers Faced by Newcomers to Open Source Projects: A Systematic Review[15]

Overcoming Open Source Project Entry Barriers with a Portal for Newcomers[17]

Preliminary Empirical Identification of Barriers Faced by Newcomers to Open Source Software Projects[21]

Social Barriers Faced by Newcomers Placing Their First Contribution in Open Source Software Projects[18]

There are many barriers newcomers face by contributing for the first time. Some barriers are larger and other are smaller. I found in my literature review 56 barriers, why newcomer may not contribute to an crowdsourcing project. I conducted barriers, which have semantically the same meaning. Also, I subdivided the barriers into X categories. This categorization and conducting the barriers will provide a more simplified and a more meaningful statement. It seems like that newcomers have the most problems to rebuild the system where the bug was in. This barrier I found most in the review

7. Support newcomer to overcome the barriers's

- Crowdsourcing in SE -[7]
 - On the Relationship between Newcomer Motivations and Contribution Barriers Open Source Projects -[3]
 - Barriers faced by newcomers to software crowdsourcing projects -[25]
 - Beyond pretty pictures: Examining the benefits of code visualization for Open Source newcomers[11]
- "results show that visualization tools made it easier for participants to find information by presenting large amounts of data in an accessible form."
- Increasing the Self-Efficacy of Newcomers to Open Source Software Projects[20]
 - Patterns for tearing down contribution barriers to FLOSS projects[24]

8. conclusion and future work

Sort the barriers and motivations and further investigate those (Es ist eine ansammlung die noch nicht unterteilt ist)

References

- [1] Enrique Estellés-Arolas and Fernando González-Ladrón-de-Guevara. "Towards an integrated crowdsourcing definition". In: *Journal of Information Science* 38.2 (2012), pp. 189–200. DOI: 10.1177/0165551512437638. eprint: <https://doi.org/10.1177/0165551512437638>. URL: <https://doi.org/10.1177/0165551512437638>.
- [2] Christoph Hannebauer, Matthias Book, and Volker Gruhn. "An Exploratory Study of Contribution Barriers Experienced by Newcomers to Open Source Software Projects". In: *Proceedings of the 1st International Workshop on CrowdSourcing in Software Engineering*. CSI-SE 2014. Hyderabad, India: ACM, 2014, pp. 11–14. ISBN: 978-1-4503-2857-9. DOI: 10.1145/2593728.2593732. URL: <http://doi.acm.org/10.1145/2593728.2593732>.
- [3] Christoph Hannebauer and Volker Gruhn. "On the Relationship Between Newcomer Motivations and Contribution Barriers in Open Source Projects". In: *Proceedings of the 13th International Symposium on Open Collaboration*. OpenSym '17. Galway, Ireland: ACM, 2017, 2:1–2:10. ISBN: 978-1-4503-5187-4. DOI: 10.1145/3125433.3125446. URL: <http://doi.acm.org/10.1145/3125433.3125446>.
- [4] M. Hossain. "Crowdsourcing: Activities, incentives and users' motivations to participate". In: *2012 International Conference on Innovation Management and Technology Research*. May 2012, pp. 501–506. DOI: 10.1109/ICIMTR.2012.6236447.
- [5] M. Hossain. "Users' motivation to participate in online crowdsourcing platforms". In: *2012 International Conference on Innovation Management and Technology Research*. May 2012, pp. 310–315. DOI: 10.1109/ICIMTR.2012.6236409.
- [6] Michael N. Huhns, Wei Li, and Wei-Tek Tsai. "Cloud-based Software Crowdsourcing (Dagstuhl Seminar 13362)". In: *Dagstuhl Reports* 3.9 (2013). Ed. by Michael N. Huhns, Wei Li, and Wei-Tek Tsai, pp. 34–58. ISSN: 2192-5283. DOI: 10.4230/DagRep.3.9.34. URL: <http://drops.dagstuhl.de/opus/volltexte/2013/4355>.
- [7] T. D. LaToza and A. van der Hoek. "Crowdsourcing in Software Engineering: Models, Motivations, and Challenges". In: *IEEE Software* 33.1 (Jan. 2016), pp. 74–80. ISSN: 0740-7459. DOI: 10.1109/MS.2016.12.
- [8] R. Leano, Z. Wang, and A. Sarma. "Labeling relevant skills in tasks: Can the crowd help?" In: *2016 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC)*. Sept. 2016, pp. 185–189. DOI: 10.1109/VLHCC.2016.7739683.
- [9] A. Lee, J. C. Carver, and A. Bosu. "Understanding the Impressions, Motivations, and Barriers of One Time Code Contributors to FLOSS Projects: A Survey". In: *2017 IEEE/ACM 39th International*

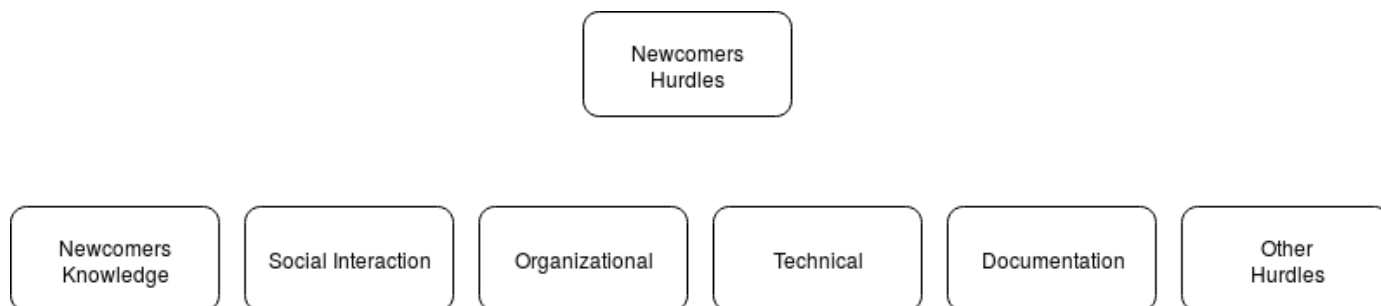


Figure 1. Categories of newcomers hurdles

- Conference on Software Engineering (ICSE)*. May 2017, pp. 187–197. DOI: 10.1109/ICSE.2017.25.
- [10] Leticia Machado et al. “The Good, the Bad and the Ugly: An Onboard Journey in Software Crowdsourcing Competitive Model”. In: *Proceedings of the 4th International Workshop on CrowdSourcing in Software Engineering*. CSI-SE ’17. Buenos Aires, Argentina: IEEE Press, 2017, pp. 2–8. ISBN: 978-1-5386-4041-8. DOI: 10.1109/CSI-SE.2017.6. URL: <https://doi.org/10.1109/CSI-SE.2017.6>.
- [11] Y. Park and C. Jensen. “Beyond pretty pictures: Examining the benefits of code visualization for Open Source newcomers”. In: *2009 5th IEEE International Workshop on Visualizing Software for Understanding and Analysis*. Sept. 2009, pp. 3–10. DOI: 10.1109/VISSOF.2009.5336433.
- [12] D. Riehle. “The Economic Motivation of Open Source Software: Stakeholder Perspectives”. In: *Computer* 40.4 (Apr. 2007), pp. 25–32. ISSN: 0018-9162. DOI: 10.1109/MC.2007.147.
- [13] Richard M. Ryan and Edward L. Deci. “Intrinsic and Extrinsic Motivations: Classic Definitions and New Directions”. In: *Contemporary Educational Psychology* 25.1 (2000), pp. 54–67. ISSN: 0361-476X. DOI: <https://doi.org/10.1006/ceps.1999.1020>. URL: <http://www.sciencedirect.com/science/article/pii/S0361476X99910202>.
- [14] I. Steinmacher, T. U. Conte, and M. A. Gerosa. “Understanding and Supporting the Choice of an Appropriate Task to Start with in Open Source Software Communities”. In: *2015 48th Hawaii International Conference on System Sciences*. Jan. 2015, pp. 5299–5308. DOI: 10.1109/HICSS.2015.624.
- [15] Igor Steinmacher, Marco Aurélio Graciotto Silva, and Marco Aurélio Gerosa. “Barriers Faced by Newcomers to Open Source Projects: A Systematic Review”. In: *Open Source Software: Mobile Open Source Technologies: 10th IFIP WG 2.13 International Conference on Open Source Systems, OSS 2014, San José, Costa Rica, May 6-9, 2014, Proceedings*. Ed. by Luis Corral et al. Berlin, Heidelberg: Springer Berlin Heidelberg, 2014, pp. 153–163. ISBN: 978-3-642-55128-4. DOI: 10.1007/978-3-642-55128-4_21. URL: https://doi.org/10.1007/978-3-642-55128-4_21.
- [16] Igor Steinmacher et al. “A systematic literature review on the barriers faced by newcomers to open source software projects”. In: *Information and Software Technology* 59.Supplement C (2015), pp. 67–85. ISSN: 0950-5849. DOI: <https://doi.org/10.1016/j.infsof.2014.11.001>. URL: <http://www.sciencedirect.com/science/article/pii/S0950584914002390>.
- [17] Igor Steinmacher et al. “Overcoming Open Source Project Entry Barriers with a Portal for Newcomers”. In: *Proceedings of the 38th International Conference on Software Engineering*. ICSE ’16. Austin, Texas: ACM, 2016, pp. 273–284. ISBN: 978-1-4503-3900-1. DOI: 10.1145/2884781.2884806. URL: <http://doi.acm.org/10.1145/2884781.2884806>.
- [18] Igor Steinmacher et al. “Social Barriers Faced by Newcomers Placing Their First Contribution in Open Source Software Projects”. In: *Proceedings of the 18th ACM Conference on Computer Supported Cooperative Work & Social Computing*. CSCW ’15. Vancouver, BC, Canada: ACM, 2015, pp. 1379–1392. ISBN: 978-1-4503-2922-4. DOI: 10.1145/2675133.2675215. URL: <http://doi.acm.org/10.1145/2675133.2675215>.
- [19] Igor Steinmacher et al. “The Hard Life of Open Source Software Project Newcomers”. In: *Proceedings of the 7th International Workshop on Cooperative and Human Aspects of Software Engineering*. CHASE 2014. Hyderabad, India: ACM, 2014, pp. 72–78. ISBN: 978-1-4503-2860-9. DOI: 10.1145/2593702.2593704. URL: <http://doi.acm.org/10.1145/2593702.2593704>.
- [20] I. Steinmacher et al. “Increasing the Self-Efficacy of Newcomers to Open Source Software Projects”. In: *2015 29th Brazilian Symposium on Software Engineering*. Sept. 2015, pp. 160–169. DOI: 10.1109/SBES.2015.10.
- [21] I. Steinmacher et al. “Preliminary Empirical Identification of Barriers Faced by Newcomers to Open Source Software Projects”. In: *2014 Brazilian Symposium on Software Engineering*. Sept. 2014, pp. 51–60. DOI: 10.1109/SBES.2014.9.
- [22] I. Steinmacher et al. “Why do newcomers abandon open source software projects?” In: *2013 6th International Workshop on Cooperative and Human Aspects of Software Engineering (CHASE)*. May 2013, pp. 25–32. DOI: 10.1109/CHASE.2013.6614728.
- [23] W. T. Tsai, W. Wu, and M. N. Huhns. “Cloud-Based Software Crowdsourcing”. In: *IEEE Internet Computing* 18.3 (May 2014), pp. 78–83. ISSN: 1089-7801. DOI: 10.1109/MIC.2014.46.

- [24] V. Wolff-Marting, C. Hannebauer, and V. Gruhn. “Patterns for tearing down contribution barriers to FLOSS projects”. In: *2013 IEEE 12th International Conference on Intelligent Software Methodologies, Tools and Techniques (SoMeT)*. Sept. 2013, pp. 9–14. DOI: 10.1109/SoMeT.2013.6645669.
- [25] A. L. Zanatta et al. “Barriers Faced by Newcomers to Software-Crowdsourcing Projects”. In: *IEEE Software* 34.2 (Mar. 2017), pp. 37–43. ISSN: 0740-7459. DOI: 10.1109/MS.2017.32.