

# The Role of Local Open Source Communities in the Development of Open Source Projects

Sinan Abdulwahhab, Yazan Alabady, Yacoub Sattar, and Imed Hammouda

Department of Computer Science and Engineering  
Chalmers and University of Gothenburg  
Gothenburg, Sweden  
`imed.hammouda@cse.gu.se`

**Abstract.** This paper investigates the position of local open source communities (LOSCs) in the development of open source projects (OSPs). We have conducted an empirical study to examine the role of LOSCs, their way of working, and the benefits/challenges they experience compared to the overall global community. The qualitative investigation consisted of ten semi-structured interviews with members within different LOSCs. The results confirm the importance of LOSCs and the pivotal role they play in the development of OSPs. In many cases, they act as the middleman between individual members and the project's global community. However, LOSCs have their own kinds of challenges.

## 1 Introduction

Open source software projects in general offer their code in an open and accessible form to the public. By being open and accessible any person with the right skills may join the community for the purpose of improving the software [1]. Developers benefit from contributing to open source projects (OSPs) in different ways including recognition from their peers, the fame they get from contributing, the advantages of the tool they create, or simply for the money [2]. In addition to developers, open source communities involve a wider set of stakeholders such as users, sponsors, and eventually business professionals [1].

The development of Open Source Projects (OSPs) is typically driven by the contributions of a geographically distributed community of people under the steering of a small core of project leaders. Yet many projects highly depend on the efforts of smaller local communities that meet face to face to work on the projects. Local communities have different purposes, some of them are educational, some promotional, some social and some are driven by financial interest. We argue that this trend has become significant and is gaining momentum through the important rise of meetups where people meet in local restricted settings to share common interests. Examples of open source communities which operate locally include Linux User Groups (LUG) and Ubuntu's LoCo teams [25, 26]. We refer to those as Local Open Source Communities (LOSCs).

The purpose of this study is to understand through an empirical investigation the way of working of LOSCs, their role and the different factors influencing their operations. In particular, the paper investigates three research questions:

- What interaction patterns are established between LOSCs and other stakeholders in open source projects?
- What is the role of LOSCs in the development of open source projects?
- What are the challenges/benefits that those communities face/bring when developing open source projects?

The study is based upon several interviews conducted with different LOSCs operating locally in different parts of the world. This paper proceeds as follows. In Section 2 we present a characterization of LOSCs. In Section 3 we discuss our research methodology. We then present our findings in Section 4 and discuss them in Section 5 in relation to existing literature. Finally, we conclude in Section 6.

## 2 Local Open Source Community (LOSC)

According to social science, a local community is a group of interacting people sharing an environment. In human communities, intent, belief, resources, preferences, needs, risks, and a number of other conditions may be present and common, affecting the identity of the participants and their degree of cohesiveness [23]. A local community is where a group of people living in a common location interact, share the same interests and contribute to each other's social or material values within a shared geographical location [23].

In open source, there are many communities that are acting locally city wise, region wise or country wise. These communities have objectives and characteristics which their members follow while working on OSPs. Examples of these communities along with their main objectives, characteristics and distribution are discussed below.

A Linux user group (LUG) is a group of developers who gather within a location and provide support, advocacy, education and a social environment for Linux users whether they are experienced or novice [15]. Furthermore, they meet face to face or via IRC to exchange information and work on various OSS projects by developing, making configurations and fixing bugs [15]. There are different characteristics a LUG can have such as a need for a website, a meeting location and a meeting time [15]. Also, LUGs are commonly known to be distributed city wise [26].

Another example is Ubuntu's LoCo teams which stand for Ubuntu's Local Community teams. They are to some extent similar to LUGs, there is a LoCo team in almost every country and sometimes more than one, like in the United States where they have it state wise [16]. The users' expertise in LoCos range from Linux experts to entirely new users [22]. LoCo teams get together to achieve objectives that include advocating Ubuntu, providing support, organizing release parties and more [22]. In order to join a LoCo team and socialize with other Ubuntu users, one has to look for a LoCo in their area, if not existent, they are allowed to start a new one with other users in the area if they are available [25].

Another example is Mozilla's Community Sites (MCS). Within MCS there are some which are locally distributed; these are commonly known as local MCS

[24]. Commonly, MCS are distributed country wise [27]. Local MCS work in a hierarchical manner and engage in various tasks such as: localization, promotion, quality assurance, documentation and extension development [24, 28]. The number of members varies among different communities, and communication mechanism varies from IRC chat to mailing lists. However, it is mandatory for local Mozilla communities to have their own specific website for information and communication mechanism [24, 28].

We refer to such initiatives as Local Open Source Communities (LOSCs). The goal of this paper is to investigate LOSCs by looking into their role in open source development, what challenges and benefits they bring or face and what interaction patterns they have with OSPs stakeholders.

### 3 Methodology

In this work, we decided to use a case study as our research strategy. A case study is an empirical inquiry that investigates a contemporary phenomenon within its real-life context, especially when the boundaries between the phenomenon and context are not clearly evident [19]. This fitted well our case given that the boundary of LOSCs in relation to individuals and global communities has not been widely studied and is not yet fully understood. With a case study, we can collect qualitative data from different developers within the local communities.

#### 3.1 Data collection procedures

Since we adopted the approach of collecting qualitative data, our aim was to conduct a number of interviews of five local communities. We have interviewed people with different roles in those communities: organizers, leaders of specialized groups, and developers. This variation helped provide different perspectives regarding the purpose of the local communities. Since the size of the sample is very small and there is no population information available, we have adopted the maximum variation sampling for both the developers and the local communities [21]. The method of data collection which we chose was the “direct method” which involves conducting interviews and the direct involvement of people [21]. We have conducted a pilot test to estimate how much time it would take to answer our questions and whether our questions were clear enough to the interviewees to provide good answers. During the interviews, different open questions were used followed by specific ones, which follows the “funnel model” [21]. With that design of data collection procedure we could achieve more solid results.

#### 3.2 The interview questions

In order not to miss any details of our research agenda, the interview questions were derived from the list of our research questions. The interviewee’s understanding of LOSC have been ensured before starting the interviews in order to prevent any misinterpretations. The list of interview questions is as follows:

1. From your experience, how do local community branches contribute to OSPs?
2. What is the difference between local branch projects and the global community projects for OSPs?
3. Why do people join LOSCs? What motivates them?
4. What are the relationships between local community members? Is it different from that between the members of a widely distributed global community?
5. How do the local communities impact the open source ecosystem?
6. How do the local communities impact the local industries?
7. What kind of challenges could a local community introduce to OSPs?
8. How can a local community benefit open source organizations and projects?
9. What benefits does the local community obtain from OSPs?
10. What challenges does the local community face when working on OSPs?

### 3.3 Method for analyzing the data

In our study, we have interviewed two developers from each of the five different LOSCs. The data we got from the interviews are mostly qualitative data. When relying primarily on qualitative data, triangulation is the best technique to use [21]. Triangulation is a data analysis technique that means studying an object from different angles and thus it provides a broader result [21]. Triangulation is also important in order to increase the precision of our research [21]. Data source triangulation is a type of triangulation where the data is collected by using more than one data source or by collecting the same data at different occasions [21].

We started with summarizing the interview data. We then sorted, organized and categorized the answers depending on which research question it is related to and who answered what. This made it easier to later analyze the data. With the sorted data, we used triangulation by analyzing what each developer answered comparing the answers in order to come up with a general result.

### 3.4 Validity threats

Although we have lots of data, there could be some threats to its validity [21]. As construct validity, the interviewed person may have interpreted our interview questions in a different way than we did. This is something which we came across in some of the interviews, specifically when we mentioned the word “local” in association with open source communities. Some of the respondents considered that all open source communities whether global or local to be the same. We addressed this concern by explaining the topic, titles and interview questions clearly and with examples for the subjects. In order to reduce this as much as possible, we had many discussions about the interview questions and we also pilot-tested them to make sure they are interpreted the same way by both the interviewer and the interviewees.

There is also the threat to the external validity. In order to minimize this, we tried to interview as many different open source communities as possible. With data from two developers each from five different well-established LOSCs, we should be able to get a generalized result.

One more validity threat is the threat to the reliability of the study. Should another researcher conduct the same study, they should, hypothetically, get the same result. In order to reduce the reliability threat, all researchers discussed the details of the research agenda to avoid possible misunderstanding among researchers and researcher bias. We also included two interviewees per case to remove possible interviewee's bias. Finally, we have our interviews recorded and transcribed not to miss any important details.

## 4 Results

We have identified three dominant organizational roles within a LOSC. First, each LOSC has an organizer, which can be seen as a local project leader. It is the organizers' work to build and grow the local community. The steps they follow to attract the developers include finding a friendly meet up place where they gather and socialize with other developers sharing the same interest. Also an organizer usually forms a mailing list or some mean of communication in order to get hold of developers for the purpose of recruiting them into the local community. The second dominant role is a developer role with tasks including development and contribution to the open source project. The third organizational element is the specialized groups within a LOSC itself. Those are smaller groups in charge of specific tasks in the local community. For instance some groups focus on very specific localization or functional features.

The discussion below is structured along four subsections each related to one of the research questions. Subsection three and four will cover the third research question. We have conducted ten interviews, three interviews with organizers of LOSCs, four interviews with local developers within a LOSC and three interviews with special groups within a LOSC. The interviewees were part of different LOSCs that are part of several open source organizations. The organizations are Linux, Meteor, Mozilla, Ubuntu and Google (buzz project).

### 4.1 First RQ: LOSC and OSP stakeholders' relationship

There were different points of view regarding different interactions between a LOSC and other OSP stakeholders. We will present the LOSCs organizers', the LOSCs developers' and the LOSC special groups' points of view.

#### *On the interaction between LOSC and the global community*

Organizers of LOSCs engage with the global community via participation in discussion boards, mailing lists and other events where members of the global community are present. LOSC developers typically contribute with what they develop within a LOSC to the project using the LOSC as medium for getting support and feedback. As reported earlier, we observed the presence of special groups within a LOSC itself. These groups support the global community with contribution around their very specific agendas.

*On the interaction between different LOSC members*

In the LOSC organizers' perspective it was emphasized that the organizers try to keep the LOSC members working on the project at hand. The members should not mix their work with their daily life's work. On another note, it was mentioned that socializing with people who use similar language/thinking is an important aspect and it makes the development more efficient.

The LOSC developers' point of view showed that the relationship between LOSC members is based on socializing factors and collaborating with others to be more effective in working.

The LOSC groups' point of view was a bit different than the others. While socializing may have been a factor, making profit was reported as an important factor. Joining to meet interesting people to work locally on profitable things is an important aspect. A group mentioned that their group was assembled at the start to work on a commercial idea, along with their contribution to the open source project.

*On the interaction between LOSC members and local industries*

The organizers' point of view was described as moral. It was said that the focus of the community should be more on issues like freedom of the software and public good when working with projects in contrary to the industry goal of making profit. Furthermore, the idea of separating LOSC and profession was enforced by the organizer and the community to remove any severe collision.

The LOSC developers' point of view seemed different from that of organizers. They mentioned that the industries could hire people from these communities. Working with open source helps being noticed by the industries. Working in LOSC increase the chance of being noticed since this activity involves a lot of interaction with the local stakeholders including the industry.

The LOSC groups' point of view was similar to that of developers. It was said that "a group of three developers I know have worked on a project and they have managed to contribute to some extent towards that project. The meteor organization has noticed their contribution and offered them a job in San Francisco".

**4.2 Second RQ: LOSC role in development of OSPs**

From the LOSC organizers' perspective, a LOSC promotes and inspires local people to use free open source technologies, as the LOSC localizes the content of open source and promotes material on a more practical level. These LOSCs work on collaborative projects and the resulting product gives benefits to both the LOSC itself and the society as a whole. Furthermore, their roles in OSP development are not limited to promoting. Various members have different roles and tasks in the LOSC as some of them provide significant code contributions to OSP.

Different LOSCs differ in their work process, however, these LOSCs share the same ethical guidelines of separating daily profession from their work on OSP, even if some of their work is in the open source field. As they contribute to these

OSPs, LOSCs use the materials and know-how of the associated open source communities.

LOSC stakeholders usually partake in discussions online and they organize and take part in international gatherings to work with other associated LOSCs from time to time on large OSFs. This grants a larger chance of solving issues in OSF development in terms of manpower. Also, this provides a larger motivational force when a large group of individuals work together, as it increases the chance of success.

According to a LOSC organizer, in order to maintain an OSF un-abandoned, it is his job to motivate developers to engage in meetups to work together and increase motivation. So in this sense, it is his job to keep the project alive locally, and possibly get more developers to contribute. The developers mostly contribute to such projects for fun and to feel part of something great. This does not disclose the fact that some projects end up being marketed for profits as the LOSC members sometimes choose to go commercial with their project by changing its license. This is because OSF can be highly beneficial and could bring large income if promoted properly. Another aspect is that some companies pay developers to join LOSCs in order to contribute to OSS platforms or tools, which the company wants to keep alive.

According to one interviewee from the Google buzz project, the role of a LOSC is to advertise and to promote the OSFs which they are contributing to. From his experience, the roles that LOSC members play in the development of OSF are not limited to coding, as promotion and marketing are big beneficial roles for OSF. For instance, members of LOSC present the project they are working on at meetups or promote it online on forums, with the purpose of attracting interest and new members. Another interview with a LOSC member from a LUG suggested the same thing, where he said that the development and success of the LOSC help the OSS movement to grow. Also, LOSC members -primarily organizers- help in structuring the OSF which the LOSC is working on in order to avoid issues in the future and to increase the chances of project success.

One of the interviewed developers made a point, as a group from his LOSC played a role in the development of an OSF and after it was released, they were still contributing by adding new packages, fixing bugs and constantly coming up with new ideas and features to the OSF. Some of his fellow LOSC members were seeking jobs, which is why they took part in the development of the OSF. This plays a big role in the expansion and spreading of OSF, as many start to contribute to them and see it as an opportunity to gain reputation which attracts job offers.

We could see the differences in the perspectives of the roles a LOSC could have in the development of an OSF. According to advertisers, promoting and marketing the project making it beneficial and profitable is what a LOSC does when taking part in an OSF. On the other hand, the developers' point of view was that LOSC helps in keeping projects alive by constantly contributing to them and adding new additions to them. From organizers perspective, LOSC

helps attract developers and keeping projects alive, and if the LOSC stopped having meetups, then the developers would start to lose motivation to work on the project and might lead them to abandoning the project they are working with, which would result in project's end.

#### **4.3 Third RQ: Benefits the LOSC bring and obtain when working in an open source project**

There are many benefits for joining LOSCs. This ranges from being a part of something greater to humanitarian reasons and enjoyment. We gathered ten data points regarding this topic, one of the biggest reason is altruism. Helping others by working with OSPs is a factor which leads many enthusiastic developers to join this cause, in a sense this practice would be a way to satisfy our nature to help.

Social factor and it's benefits encourage many to join. It helps developers to grow and work in an environment which one can get an indication on how the industry operates. Other benefits include sharing ideas within the LOSC, being able to discover new ideas and create them. This leads to improving one's business relationships and goal orientation.

The last and probably the biggest motivation was experience and creating opportunities for future work. Many work with open source to get some recognition. It could be through the open source product itself or popularity from code repositories. Several interviewees clearly stated they received many job offers due to their contribution and local recognition from OSPs which they have worked on. This has been reported to be the biggest beneficial factor of joining LOSCs and OSPs.

#### **4.4 Fourth RQ: Challenges which the LOSC bring/face when working in an open source project**

From the organizer point of view, the main challenges were limited funding and tool changes. Due to organizers focus on promotion and the nature of open source, economical cost was a challenge often faced. Another challenge was the fact that the results may not meet the goals set for a code distribution. Another challenge that a LoCo organizer mentioned was that the LOSC scene is dying since many developers are working directly within the global communities instead of joining LOSC as he sees it. Lastly tool changes are a challenge faced. The challenge revolves around how to introduce new tools to a LOSC. This could lead to structural challenges and efficiency problems with project development.

From a LOSC group point of view, the challenges faced by groups are more general. The biggest challenges usually include meeting up with the group since time and place could be an issue as most of the members work in their spare time and individuals have different schedules based on their separate life styles.

From one LOSC member point of view, the challenge which LOSC members face is adaption, whenever there is a new tool introduced or when faced with structural changes. The majority of members in a LOSC are developers, and

whenever there is a new tool introduced many try to reject the idea at first. This is because when they are required to use the tool, the adaption process slows down the development process and some at the start might be too conservative to learn it. This leads to slowing down not only the process but also the workforce. Another issue is related to structural changes. According to some of our interviewees when a structural change occurs the members of the community tend to get the shorter stick. They have to adapt, learn and understand the changes which sometimes lower the morale of the members.

## 5 Discussion

In this section we use our findings to discuss the importance of LOSCs, the main motivations to join them and the challenges faced.

### 5.1 On the importance of LOSCs

We have found out a number of important aspects when it comes to the importance of the LOSCs.

#### *Collaborative work*

The collaborative work which LOSCs provide is quite important. While working on an OSS project there are tools which allow the developers to work collaboratively. These tools are for example the discussion boards which help developers get more support or tools like GitHub which helps them organize their contributions. Those tools are accessible to the LOSC members. However, talking face to face, socializing and working in a team together in the same place is more collaborative and more efficient. Furthermore, joining a LOSC gives developers coding experience and social industrial experience by working in an office like environment and collaborating with developers of various skill sets.

#### *Meeting people with the same views*

Within a LOSC you could meet people who share the same views and goals as you. Maybe those people you meet could be your future teammates who work with you on something you both are interested in. We found out such relationships within the groups in the LOSCs. Such relations are hard to run into in the open source communities since global interactions are mainly in the cyber world.

#### *Interaction with industry*

Many global community developers try to earn respect and good reputation in the OSS community by contributing with code [10]. This is a common way for a global community developer to be noticed by the industry. LOSC introduces a new way for developers to be noticed by the industry. By meeting people from the industry at a LOSC, members within that LOSC will have a higher chance of being hired at industries since their work, skills and contributions might get noticed by those people. This indicates that a LOSC member has a higher chance

of being noticed by industries compared to a global community developer.

#### *Promoting open source*

On a different point of view, the LOSC could be considered as a mean to inspire people to use free open source technologies. The meetings and socializing aspects along with the support provided by the experts within the LOSC could assist in inspiring people to use and contribute to the open source. The LOSC can also be used as a way to promote different OSPs and different open source tools.

### **5.2 Why people join LOSCs**

Open source community's work force is to a large extent based upon volunteers. The majority of these volunteers work for free and on their spare time but what is the golden factor which makes them join?

There were several factors in the literature findings such as motivation to learn and create, social motivators, flow motivators and altruism motivators. Those motivators helped open source community developers to contribute to OSPs [9]. Other motivations include intrinsic factors and extrinsic factors [4, 6, 13]. From our interviewees we had factors such as social reasons, future work criterion, enjoyment, self-improvement and humanitarian. If we would compare these factors from both sides we can clearly see all of them match each other, even though the wording is different.

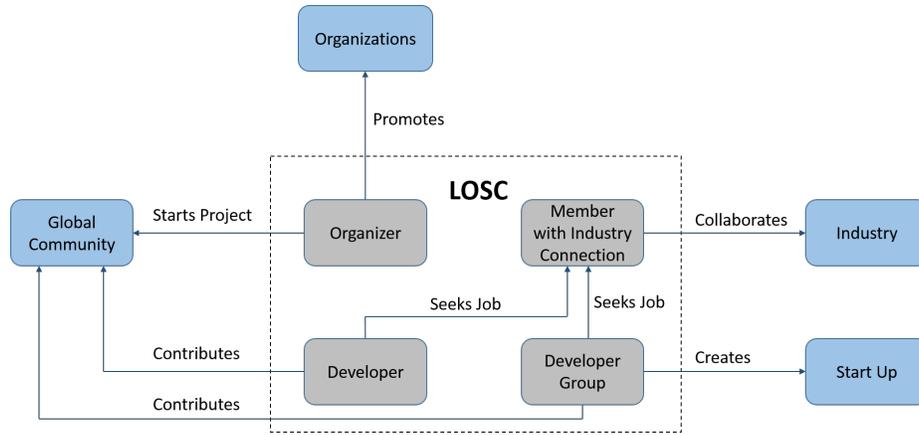
### **5.3 Common challenges associated with LOSCs**

While working with open source or any kind of work in general anomalies might emerge in forms of challenges; the nature of the challenge differs when comparing literature and our gathered data points. According to the gathered challenges from the literature the three biggest challenges are; low level activity and performance, lack of documentation and support roles, and the forking of projects and high-end users' product development [11]. Another very common challenge that many would face is the lack of focus on documentation and support, the need of a decent user interface and backward compatibility which is found in different OSPs [11]. From our interviews we gathered three different challenges from different views. First view was organizers' point of view where the concern revolved around promotion of LOSC and tool introduction to the LOSC. Second, local groups' view is the issue of time and date. Third, LOSC members' view, organizational and structure changes.

### **5.4 A conceptual model**

Figure 1 depicts an overall conceptual model for LOSCs as described in the previous discussion. There are both internal and external stakeholders that regulate or influence the working of LOSCs. In addition, the figure highlights the different interactions and relationships between those stakeholders. In order for LOSCs

to operate sustainably, it is important to understand the expectations of each stakeholder in terms of what it receives and what it provides. However, because the proposed conceptual model explains the phenomenon of LOSCs applied to a limited number of cases, the conceptual model needs to be tested further and linked to theoretical studies in related fields such as social sciences.



**Fig. 1.** Conceptual Model for LOSCs.

## 6 Conclusions

We conducted this work with the aim to understand the role of LOSCs in the development of open source projects. In order to achieve this we have chosen a case study strategy as our research methodology. To investigate our topic in-depth, we conducted ten interviews with members of several LOSCs, who had different roles within those local groups. From our results we concluded that both LOSCs and OSPs do impact each other. Some impacts are beneficial such as code contributions, project localization and promoting within the local region. Others represented challenges such as financial challenges and project abandons.

We have also discussed motivations as to why people join LOSCs rather than being in contact solely with the global community. Motivations included the need of being part of something big while working locally, helping humanity by offering free software and socializing with people who share the same interests.

A total of ten interviews have been conducted in order to investigate LOSCs. The study should therefore be classified as a preliminary inquiry, as opposed to a substantial research which examines the OSS community dynamics over time across the various local OSS projects. Our future work consists of studying factors of success and failure of LOSCs and how to build and grow sustainable LOSCs. We also plan to widen our study to other open source projects and compare between different LOSCs within the same project.

## References

1. Bonaccorsi, A., and Rossi, C. (2003). Why open source software can succeed. *Research policy*, 32(7), 1243-1258.
2. Godfrey, Michael W., and Qiang Tu. "Evolution in open source software: A case study." *Software Maintenance*, 2000. Proceedings. International Conference on. IEEE, 2000.
3. Dahlander, L., and Magnusson, M. G. (2005). Relationships between open source software companies and communities: Observations from Nordic firms. *Research policy*, 34(4), 481-493.
4. Wang, F. R., He, D., and Chen, J. (2005, August). Motivations of individuals and firms participating in open source community. In *Machine Learning and Cybernetics*, 2005. Proceedings of 2005 International Conference on (Vol. 1, pp. 309-314). IEEE.
5. Wang, Y., and Guo, D. (2007). EMOS/1: An Evolution Metrics Model for Open Source Software. Unpublished Paper.
6. Deci, E. L., Koestner, R., and Ryan, R. M. (1999). A meta-analytic review of experiments examining the effects of extrinsic rewards on intrinsic motivation. *Psychological bulletin*, 125(6), 627.
7. den Besten, M., Dalle, J. M., and Galia, F. (2008). The allocation of collaborative efforts in open-source software. *Information Economics and Policy*, 20(4), 316-322.
8. Ghapanchi, A. H., and Aurum, A. (2012). The impact of project capabilities on project performance: Case of open source software projects. *International Journal of Project Management*, 30(4), 407-417.
9. Baytiyeh, H., and Pfaffman, J. (2010). Open source software: A community of altruists. *Computers in Human Behavior*, 26(6), 1345-1354.
10. Feller, J. (2005). *Perspectives on free and open source software*. MIT Press.
11. Lerner, J., and Triole, J. (2000). *The simple economics of open source* (No. w7600). National Bureau of Economic Research.
12. DeKoenigsberg, G. (2008, April). How successful open source projects work, and how and why to introduce students to the open source world. In *Software Engineering Education and Training*, 2008. CSEET'08. IEEE 21st Conference on (pp. 274-276). IEEE.
13. Lakhani, K., and Wolf, R. G. (2003). Why hackers do what they do: Understanding motivation and effort in free/open source software projects.
14. Dick, S., and Sadia, A. (2006, July). Fuzzy clustering of open-source software quality data: A case study of Mozilla. In *Neural Networks*, 2006. IJCNN'06. International Joint Conference on (pp. 4089-4096). IEEE.
15. Moen, R. (2004). *Linux user group HOWTO*.
16. *Ubuntu LoCo Teams List* — Ubuntu LoCo Team Portal. (n.d.). Retrieved December 13, 2015, from <http://loco.ubuntu.com/teams/>
17. Raymond, E. (1999). The cathedral and the bazaar. *Knowledge, Technology and Policy*, 12(3), 23-49.
18. Mockus, A., Fielding, R. T., and Herbsleb, J. D. (2002). Two case studies of open source software development: Apache and Mozilla. *ACM Transactions on Software Engineering and Methodology (TOSEM)*, 11(3), 309-346.
19. Yin, R. K. (2013). *Case study research: Design and methods*. Sage publications.
20. Creswell, John W. *Research design: Qualitative, quantitative, and mixed methods approaches*. Sage publications, 2013.

21. Runeson, P, Hst, M (2009) Guidelines for conducting and reporting case study research in software engineering, *Empirical Software Engineering*, vol 14, pp. 131-164.
22. Bacon, Jono. *The art of community: Building the new age of participation.* ” O’Reilly Media, Inc.”, 2012.
23. Beck, U. (1992). *Risk society: Towards a new modernity* (Vol. 17). Sage.
24. Mozilla Community Sites. (n.d.). Retrieved December 13, 2015, from <https://wiki.mozilla.org/MCS>
25. Ubuntu Wiki. (n.d.). Retrieved December 13, 2015, from <https://wiki.ubuntu.com/LoCoFAQ>
26. UK Linux User Groups. (n.d.). Retrieved December 13, 2015, from <https://lug.org.uk/uklugs>
27. Contacts, Spaces and Communities - Communities. (n.d.). Retrieved December 13, 2015, from <https://www.mozilla.org/en-US/contact/communities/>
28. MCS/Planning. (n.d.). Retrieved December 13, 2015, from <https://wiki.mozilla.org/MCS/Planning>