Open Source Software as a Learning Tool for Computer Science Students

Assiya Khuzyakhmetova and Aidarbek Suleimenov

Nazarbayev University, Astana, Kazakhstan
{aidarbek.suleimenov,assiya.khuzyakhmetova}@nu.edu.kz

Abstract. In this paper authors' experience of contributing to Open Source Software (OSS) is described. Contributions were done as a part of the OSS course taken at Nazarbayev University during the Spring 2019 term. Two junior bachelors degree students described their experience, motivations to contribute to OSS, selected projects, course structure and the lists of activities they performed. Assessment of this experience by other community members and the course instructor is also mentioned in this publication. This paper also studies how the course structure can affect people's ability to make contributions in general.

Keywords: Open Source Software · Student experience · Distributed development · Computer Science Learning

1 Introduction

This paper describes the authors' experience in contribution to Open Source Software (OSS). Although there are differences in terms of background, motivations, and expectations, both of junior bachelors degree students took OSS course in Nazarbayev University during the Spring 2019 term. The structure of the course allowed flexibility in terms of project selection and types of contributions with the only requirement of being fluent with the programming language of the project. Students were allowed to freely choose any OSS project in any language they are proficient with. Therefore, this paper tries to evaluate if freedom of choice in terms of OSS project and technology affects students' ability to contribute. One example of a similar approach took place in University of Skövde [1]. This study shows that masters students successfully made contributions to OSS projects as a part of the course. However, the question if the same approach could be applied to bachelors degree students remains open. This paper aims to investigate this issue and proof or disproof if freedom of choice in OSS course affects the outcome. This is done by evaluating two chosen students' experience in contributing to OSS projects. The individual experience of each student is described and they are referred to as Student A and Student B respectively next in this paper. Generalized conclusion of both students' work and a piece of advice for future contributors are presented in the paper.

1.1 Course structure

One part of the OSS course included information about the types of communities, governance structures, licenses, and other theoretical topics about OSS. Another part included project selection, interaction with the community supporting the chosen project, work on the project, and presentations of our contributions to the class and professor. Speaking about the flexibility of roles, students could choose between roles of OSS contributor, OSS observer, and OSS consultant, and each of the roles had their assessment criteria. Both of these students have chosen OSS contributor roles, specifically being Code Developers. Although both of them were junior Computer Science students they had different programming background that was influenced by previous experience in Competitive Programming, having different courses in university and different prior work experience. This resulted in different challenges that they faced during the course, i.e. Student A had challenges in learning new technologies, while Student B had mostly challenges related to interaction with the community.

1.2 Students' motivations

Both of the students also had the motivation to learn new technologies and things during the period of the course through contributions to OSS. Student A wanted to learn more about Machine Learning, whereas Student B wanted to learn how to use different databases. Moreover, Student B also wanted to improve his career prospects through OSS contributions. Speaking about their intrinsic motivation, they are interested in Open Source and they wanted to feel like a part of a big team of Open Source Developers doing something useful for the society. This resulted in Student A choosing "Mozilla BugBug" project and Student B choosing "Redash" project. Descriptions of these projects will be described in this paper. Moreover, the following paper will combine their experience of contributing to these OSS projects and will describe their motivations, contributions, and challenges that they faced more closely.

2 Contributed OSS projects Description

2.1 Mozilla BugBug

Mozilla BugBug project is a Platform for Bugzilla Machine Learning projects. Bugzilla is server software designed to help manage software development [4]. The project aims to apply Machine Learning models to Bugzilla Dataset with bugs to learn it to automatically detect such thing as types of bugs, to identify whether a bug needs quality assurance, or to automatically assign a person to the bug, etc. This project, like any other Machine Learning project, heavily relies on data collection, and for this purpose, there is a Bugzilla Data Collector project written in JavaScript as a Firefox Web Extension. Since Student A wanted to learn Machine Learning, BugBug project was chosen because of its technical aspects.

Technical Aspects of the project. Mozilla BugBug project is written in Python and XGBoost optimized distributed gradient boosting library is used as a classifier but any other classifier can be applied. The project contains data processing pipelines that use several Mozilla APIs to feed the data to the classifier. Training is performed with the neural-network library. Simple JSON database implemented for data storage. Taskcluster with Docker is used for building and testing of the project. Dependency updates are done by PyUp bot [10]. Firefox Web Extension[3] is used primarily for data extraction. It is written in JavaScript with NodeJS framework and HTML. Collected data is saved as CSV file. Open issues that are available for contributors are under the issues tab of the projects. There are currently 104 issues in BugBug [6] and 4 in Bugzilla Data Collector [7].

General Aspects of the project and community. This particular project belongs to the Mozilla Foundation. While Mozilla can be considered to have Federal Leadership model, BugBug has Monarchical Leadership model since the main decisions are made by one person. The tasks allocation process is voluntary. Specifically, tasks are freely selected by participants. Issues are not assigned to any of the contributors until there is a Pull Request referring to this issue. Code owner is the same person as a Leader of the Community. The license that this project has is MPL (Mozilla Public License)

2.2 Redash Project

As todays world is becoming more and more data-driven, it is important for companies and governments to analyze data that they have and to easily extract some valuable insights from them. There are many solutions to this problem, but most of them are proprietary and expensive (like Tableau). On the contrary, Redash [11] is an open-source solution, which helps to democratize the data analysis and visualization and makes it available for all. For this reason, Student B found the projects mission quite appealing.

Technical Aspects of the project. The project itself is a single repository on Github hosting provider [11]. Currently, the repository consists of 2 server applications: one is the API written in Python programming language with Flask framework and another one is a server dedicated to distributing code for front end application written in Angular/ReactJS. Project data is stored in the PostgreSQL database and project uses Docker image for simplifying hosting and testing. There are currently more than 300 different issues raised in the areas of the front end, back end, UI, UX, etc. 2 big improvements are being made to the project. First is complete migration from Angular to a ReactJS and second one is to redo of the permission system.

General Aspects of the project. The leader of the project is the CEO of the company with the same name, which was established on top of the open-

source software. In theory, all critical decisions regarding the project could be made by company employees. However, the number of employees is so small that governance structure can be considered as monarchical, which essentially means that all decisions are made by the CEO. Most of the big decisions are proposed by him, but anyone can make some suggestions or raise issues. Several people are the core members of the project, but not all of them work in the Redash company. Some of the core members are working at Mozilla or the biggest Russian sports website, while others still studying at university. The typical contribution path is going from looking for issues on the GitHub page, forking the project, committing changes and opening a pull request. The Open Source software is distributed under the BSD-2 license.

3 Contribution

3.1 Contributions to Mozilla BugBug project

Project selection process. Motivation is a significant factor to consider when speaking about the project selection process. Student A motivation could have been divided into 4 main kinds: personal intrinsic, social intrinsic, technical extrinsic and economic extrinsic [12]. Technical extrinsic motivation had the most important impact on her choice since she wanted to gain experience in Machine Learning and improve existing skills in Python, and Mozilla BugBug project was the best candidate for this. During Spring term Student A accomplished 2 tutorials on Machine Learning and learned how to add a feature, what does labeling of data means and how accuracy can be changed from the addition of wrong features. This also made her interested in Machine Learning and she started Coursera Stanford course on Machine Learning upon completion of Spring term. She has also learned how to work with web extensions for the Firefox browser and how to collect data using them.

Role in the selected project. Student A has chosen a Contributor role, specifically, a Code Developer. The project can be considered active since it has many closed Pull Requests in the last month and the average response time is less than 1 day during the workweek. Also, due to the number of commits to the repository of BugBug (4 commits), on the end date of Spring term (April 14) Student A was the 7th contributor in the project out of 24.

Interaction with the community. Student A started an interaction with the community on January 22 by adding an issue related to a bug that she found in Bugzilla Data Collector. After fixing that issue, Student A started communication with BugBug project and her contributions described in the next section.

Fig. 1. Initial interaction with the community[5].

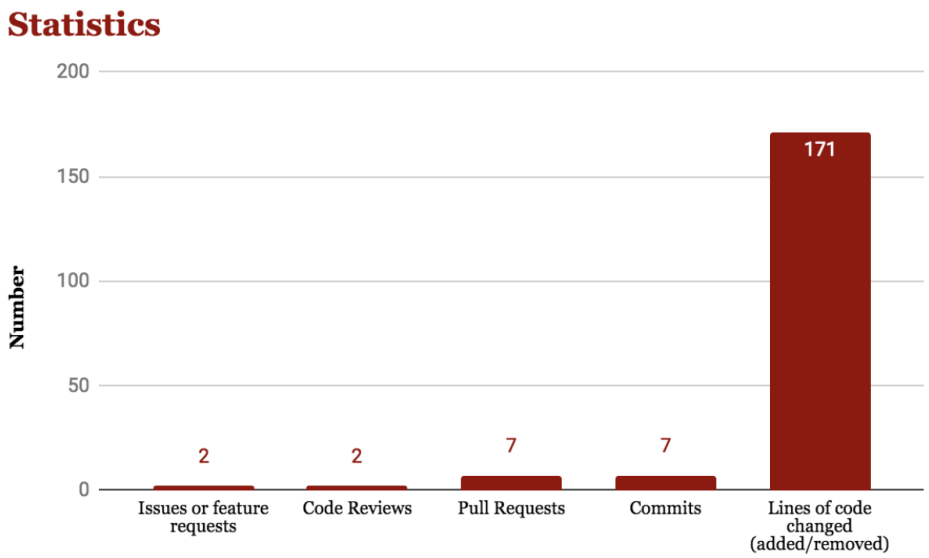

Fig. 2. Statistics on Student A activity during the Spring Term.

Activities and challenges. Student A's contributions to the project could be divided into 3 parts: code commits, code reviews, and creation of issues or feature requests. Statistics on contributions in terms of issues or feature requests, code reviews, pull requests, commits and number of lines of code changed can be found in Figure 2 below. Speaking about the challenges that were faced, it was challenging to contribute to the project that used technologies that were not used before.

3.2 Contributions to Redash project

Project selection process. The main motivation of Student B for choosing the project was to acquire new skills, learn new technologies and to improve his employment opportunities. On the other hand, the project should have used programming languages that he already knew, since otherwise, his contributions wouldn't be valuable enough. Therefore, there always should be a good balance between what he already knew and what he would like to learn. Firstly, Student B was looking for projects in **Go** programming language, since he wanted to learn more about it. However, he couldn't find an appropriate project with this programming language, because he was too novice in using it. That's why Student B decided to find a project in **Python** programming language with which he was more familiar. Regarding improving his employment opportunities, Student B wanted the project to be popular enough, but not too popular since it's hard to make contributions with many community members. Redash has more than 13 000 "stars" on the Github page and widely used in different organizations [11]. For that reason, Student B had chosen Redash as the selected project.

Role in the selected project. Student B chose the Redash project and decided to be a contributor there. More specifically, he decided to be Code Contributor. Firstly, Student B planned to look at the issues of the project from the Github page and search for the ones he can solve. The project is very popular with more than 300 issues open [8], which makes it easy enough to find different types of problems. He planned to solve as many simple issues as possible so that Student B could become more comfortable in solving harder ones.

Interaction with the community. Student B's community interaction was done asynchronously and took place during discussions of certain issues on the Github issues page [8]. Mainly it was asking for clarifications regarding the issues. There were also some discussions in the comments to the Pull Requests page as well [9].

Activities and challenges. During the course, Student B made 3 pull requests accepted to the main projects repository and 2 were waiting to be accepted. Moreover, he also made one contribution once the course was finished. Almost all

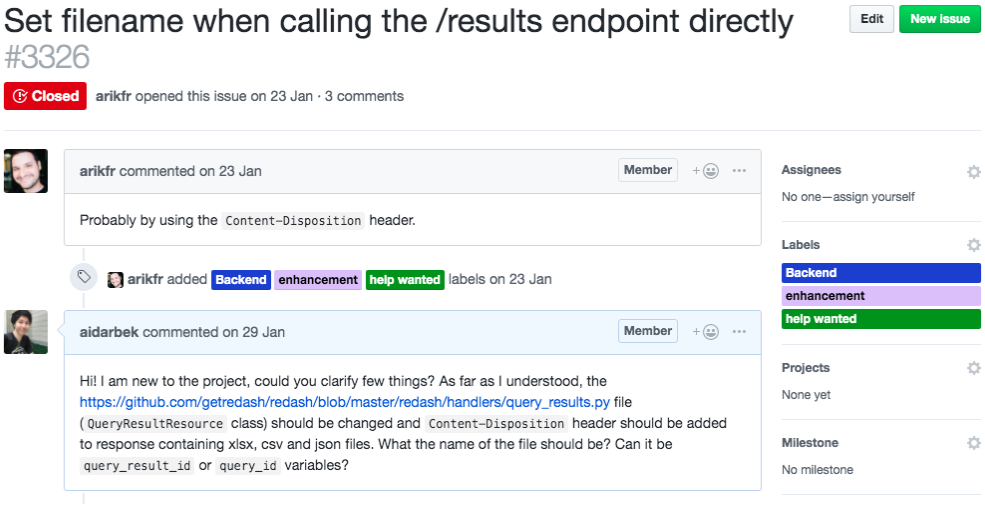

Fig. 3. Example of student B' clarifications to the issue

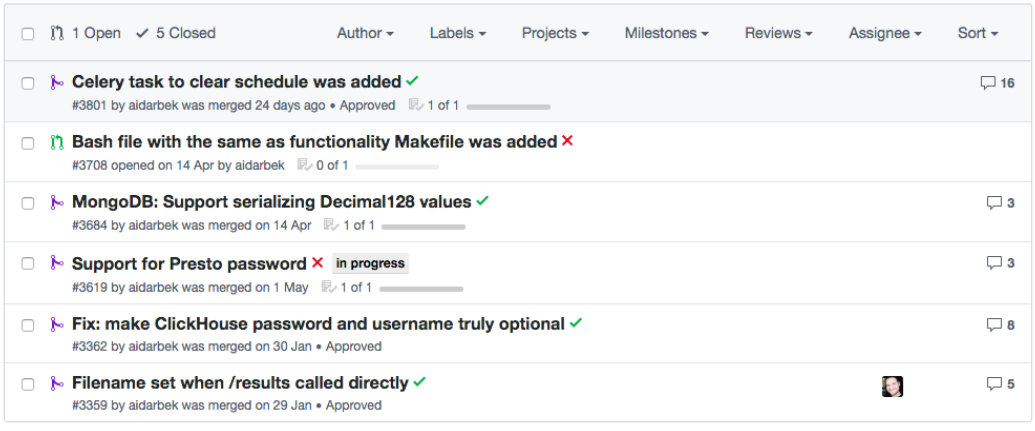

Fig. 4. List of student B's Pull Requests (closed and open)

of these were made using Python programming language and one was made using Bash. Scope of his code heavily varied: adding HTTP headers to the response, MongoDB type error fixes, Clickhouse and Presto settings configuration changes, creation of Celery tasks and Bash script writing. The main reason for choosing such a different type of tasks was to learn how to use new technologies. It was also a challenging thing to do, as each issue turned out to be related to a different technology. Apart from technologies, the scope of work also differed completely. Some issues took only a couple of hours to solve with more than a hundred lines of code, while others required to spend up to one week for them resulting only in three lines of code. The community itself was very helpful in solving different issues as well as providing feedback to solutions. As a result of Student B's interaction and contributions, he was invited to become maintainer in the Redash community. The role of the maintainer is rather symbolic but has some additional privileges like labeling certain issues, assigning people to review and accepting other Pull Requests. Apart from Redash, Student B also contributed to Beats [2] project after the course finished. The project and contributions were written in **Go** programming language, which he was eager to learn at the beginning of the course. His experience with the Redash project helped to increase confidence and try more challenging tasks and also learn a new language.

4 Conclusion and future work.

4.1 Students' conclusion

Contribution to Open Source projects can be considered as a powerful learning tool for students that allows to learn and practice technical skills [13]. Both students indicated that the experience of contributing to Open Source Software was very helpful for the professional development and helped to understand how big projects are done, how to interact with OSS communities, create contributions and overcome technical obstacles. Both students were happy with the choice of projects they made and the results of the work done during the course. As was noted by Student B, the things that were learned during contribution, according to his experience, then were easily transferred when contributing to completely different OSS projects. Speaking about differences in experiences, there were delays in communication with the community from the side of Student B, while it was not the case at Student A. The reason for that is that Mozilla's community has instant messaging communication IRC channel IRC in Mozilla IRC Cloud in contrast to Redash, where communication was mostly done on GitHub. For that reason, Student B needed to ask for a response of community core members several times. This happened several times, however, it was not the case in most of the interactions. The possible reason for this is the high volume of work taken by the main creator of the project. This created some inconveniences that are usually absent at the regular software engineering job. As for advice from both students to newbie contributors, it is significant to find a middle between the desire to learn new technologies relevant to the project and being able to be useful in the project with your current knowledge. This could prevent many

difficulties in contributing to Open Source Software. One more thing is not to be afraid of asking questions and making mistakes. Usually, a contributor can simply ask for help in the comments under issues or Pull Requests and to get help in a short period.

4.2 Study conclusion

Introduction of the course of Open Source Software to the curriculum of universities and colleges could benefit many students in both short-term and long-term perspectives boosting their CVs, technical and social skills. Furthermore, the freedom of choice in terms of projects to contribute and technologies to use has proven to result in successful OSS contributions, which supports current studies [1]. The success of contributions is assessed by OSS communities' members as each piece of code passed a rigorous assessment by several people. Speaking about assessment from the course side, both students received the highest possible mark by the course instructor.

4.3 Future work.

The future work in this research could include analysis of the experience of a higher number of students in OSS. Moreover, feedback and survey of the core members of the OSS community could be included.

References

1. Lundell, B., Persson, A., Lings, B.: Learning through practical involvement in the OSS ecosystem: Experiences from a masters assignment. In: Feller, J., Fitzgerald, B., Scacchi, W., Sillitti, A. (eds.) Open Source Development, Adoption and Innovation. pp. 289294. Springer US, Boston, MA (2007)
2. Beats Repository, https://github.com/elastic/beats
3. Bugzilla Data Collector Repository, https://github.com/marco-c/bugzilla-data-collector
4. Bugzilla Official Site, https://www.bugzilla.org/
5. GitHub Issue 5 in Bugzilla Data Collector, https://github.com/marco-c/bugzilla-data-collector/issues/5. Last accessed 6 July 2019
6. List of Issues in BugBug project, https://github.com/mozilla/bugbug/issues. Last accessed 6 July 2019
7. List of Issues in Bugzilla Data Collector project, https://github.com/marco-c/bugzilla-data-collector/issues. Last accessed 7 July 2019
8. List of Issues in Redash project, https://github.com/getredash/redash/issues. Last accessed 7 July 2019
9. List of Pull Requests in Redash project, https://github.com/getredash/redash/pulls. Last accessed 7 July 2019
10. PyUp Bot Repository, https://github.com/pyup-bot
11. Redash Repository, https://github.com/getredash/redash
12. S. Androutsellis-Theotokis and D. Spinellis and M. Kechagia and G. Gousios. (2010). Open Source Software: A Survey from 10,000 Feet.
13. Kuk, George. (2006). Strategic Interaction and Knowledge Sharing in the KDE Developer Mailing List. https://doi.org/10.1287/mnsc.1060.0551