# Practical Course – Contributing to an Open-Source Project

# Final Report

Helena Klause

Winter term 2020/21

Zulip has been a great project to work on. The setup of the development environment is easy and fast, the quality of the documentation stellar, the response times low and the atmosphere in the community especially welcoming. I have gained a lot of technical knowledge during the last months of contributing, and gotten an insider look at how the project is run. Zulip's open-source philosophy, the perspective of a core developer and Tim Abbott's role in the project have been particularly interesting.

## 1 Project summary

Zulip is a full-featured open source group chat that originated form the start-up Zulip Inc. founded in 2011 by Tim Abbott and three other founders. The company was acquired by Dropbox, but was not developed further until September 2016 when Dropbox agreed to open-source it. Zulip became practically independent of Dropbox and quickly developed a well-functioning community of developers. Abbott started a company called Kandra Labs in order to staff a core team of developers working full-time on the project, and is still the main project lead to this day.[1]

The motivation behind the project is to develop a group chat that is optimized for productivity. Many companies, FLOSS projects and other kind of groups use Zulip and value its unique features, e.g. having a topic for every stream message.[1]

Zulip is financed by a grant from the US National Science Foundation, Tim Abbot's savings, and the paid plans for hosting the software and prioritized support.[1]

# 2 Tackled issues

## 2.1 Improve the behavior of hotspots: deep dive

Zulip's community chat offers a stream (channel) for all newcomers to introduce themselves. I used this opportunity and wrote a few words about myself. One of the core developers (Vishnu KS) replied quickly by recommending a good first issue to me. Happy about the smooth first encounter with the community, I started working on the issue right away. The issue was not tracked on GitHub so far but only consisted of a topic inside the chat.[2] Vishnu confirmed that there is no need to create one, since I will start working on it soon.

The issue aims to improve the user experience for people that are already familiar with Zulip. Small pop-ups are used to introduce different features to new users to ensure an easy start. Since many people do not use Zulip only for one organization but for multiple, they might get these pop-ups, also called hotspots, every time they join a new group chat. The goal was to implement an improvement to only show the hotspots once for user profiles that are registered with the same email address. This eliminates the annoyance of having to click through the hotspots again once the user has seen them.

Since the general task was clear to me and I already made myself familiar with the respective codebase I started right away with my implementation. During my work I found the subsystem documentation for hotspots which helped me improve my understanding of the code.[3] I also added new tests and altered existing one to fit the new behavior.

After I pushed my commits to master, Vishnu reviewed them after only one day.[4] He pointed out different possible approaches and described their advantages and disadvantages. I felt grateful for his long explanations and changed my implementation to fit his suggestions. One learning I took out of this is to discuss the approach in more detail before starting to write code.

Vishnu and I had some back and forth on my PR for around two and a half weeks and he always made me feel like my work is appreciated and useful. He also communicated clearly and meticulously. I learned a lot about how code review is done in practice. The PR was now ready to be merged by Tim Abbott, which might take a while since

he had been away from the project for some weeks at that point. Vishnu offered me to ask him to prioritize my PR, but I indicated that I was happy to wait (Vishnu KS, personal communication, February 15, 2021). Later on I had to resolve some conflicts with the master branch that appeared on my PR. I also found an old GitHub issue that is partially addressed by my changes and left a comment to link my PR.[5]

## 2.2 Custom emojis do not show up in search

A user reported a bug where custom emojis were not showing up in search, and I started working on a reproduction.[6] After investigating the behavior on a local development server, I could not reproduce the bug and realized that it must have been solved in a commit between master and the latest release. I reported my findings and dropped the issue. While working on this, I found a typo in the documentation and fixed it with a quick PR.[7]

## 2.3 Welcome emails

Zulip sends the same welcome email to all users when they join a group chat, even if they have already signed up with the same email to another chat. There was a two-year-old GitHub issue for slimming down the welcome email for returning users.[5] After discussing the current status of the issue with a core developer who concluded that it is still a desirable change, I started working on it and made a pull request.[8]

Tim Abbott closed the PR and issue, explaining that he is happy with the current implementation. As the lead developer he always has the last word in any decisions. It would be great if Zulip would improve the system of tidying up old issues more frequently, to avoid open issues that include undesired changes. Since this PR was not too much effort it did not affect my motivation negatively that it was closed right away.

## 2.4 Invite system

I found a backend issue created by Vishnu KS concerning the invite system.[9] Having learned from my mistake, I explicitly asked Vishnu KS and Tim Abbott if the requested changes are still desired. While waiting for an answer, I already made myself familiar with the relevant part of the code. Vishnu replied, confirming that the issue is still current and linking a chat conversation. Tim Abbott never replied since he was off work at that time, but I read in an old converstation in the chat that he is not sure wether

the suggested approach is useful and also considers it as low priority.[10] This made me decide to not work on the issue and try to find a different one that has a higher priority.

## 2.5 Custom hotkeys

I found an issue to add a new feature that makes it possible to define custom hotkeys, which seemed to be an interesting technical challenge for me.[11] Even though the original GitHub issue dates back to 2017, I found other issues ([12][13]) that were referring to the same problem, one of which was more recent. So I decided that the feature request is still current and claimed the issue. This is a significant new feature that requires changes in a lot of different parts in the codebase. I did an extensive study of possible solutions and outlined a few alternative designs. I made sure to include answers to all design questions that were mentioned in the issue description.[14]

Shortly after I posted my approach, two core developers of Zulip answered that they do not want the feature anymore and suggested implementing support for hotkeys on all international keyboards instead of adding the ability to customize hotkeys in general.[13] I am still happy about spending that time and effort because I gained a lot of new knowledge, and because this issue was closed, which is also progress. While I was reading the relevant documentation I found an incorrect sentence and corrected it.[15]

## 2.6 Enable hotkeys on any keyboard layout

Compared to implementing custom hotkeys, which involved backend and frontend, making hotkeys work on any international keyboard layout is only a frontend issue.[13] I had close to no experience in frontend development but felt ready for the challenge. It took some time to get familiar with Javascript and the frontend logic for hotkeys. I started a discussion in the chat about different approaches, the current state of technologies, and conclusions from testing. I iterated on a few implementations to see how they work out in practice and if they actually solve the problem. During the whole time, I got feedback in the chat from different core developers. The responses were always friendly and helpful. Nevertheless, I did not get to make a PR since I could not find an implementation that fulfills the requirements without major downsides. I would like to continue working on this issue in the future.

# 3 General reflection

Contributing to Zulip has been my first interaction with any FLOSS community. Especially in the beginning, it was a totally new feeling for me to have discussions in public and write code that is visible to everyone. I reacted to this with a self-imposed perfectionism that blocked me in many ways. I repeatedly spent a lot of time overthinking the same small details which made my work unproductive and did not even help me to increase the quality of my comments or code. This behavior of mine was partially triggered by the extensive documentation, e.g. the very detailed and strict commit discipline that made me feel like I would surely make a mistake.[16] I realized this pattern in my work rather early and started to time-box specific tasks and also set precise goals for the day. It also helped to see how friendly the replies to all my messages were.

On the other side, Zulip's documentation has also helped me a lot.[17] Many design decision are well documented and a lot of subsystem sections help to get familiar with new parts of the code. The general quality is very high and written in an understandable way.

Finding recently created issues that are ready to work on appeared to be rather difficult for me. Suitable issues were rapidly claimed by other contributors, so I found myself going through issues that were already a few years old. It was challenging for me to start working on issues that are not well-described, that do not fully fit my skills or that require changes in parts of the code that I have never touched before. I think that this got significantly better from week to week, but I would still give myself the advice to be even more proactive, since Zulip's community is quite responsive and happy to help with any kind of problem.

Overall Zulip has been a great project to work on. I would likely choose it again as project to contribute to and would also recommend it to others.

# 4 Communication with the community

During my work, I experienced how welcoming and friendly Zulip's community is. I have not seen any aggressive or disrespectful messages, neither by new contributors nor by full-time developers. This reflects my impression of the project's internal communication in the preliminary report.[1] Furthermore, response times are exceptionally short. I received an answer to all my development oriented messages in the chat within an average of two hours.[18] Considering that many people live in a different timezone than CET and that

some of the questions required specialized knowledge, this is a great result.

Many questions that I came across were not even necessary to ask because they were already answered in the chat and easy to find using Zulip's extended search options. This is likely a result of the high traffic in Zulip's community chat. In the week from March 9 to March 15 in 2021 there were an average of 200 messages a day for the default streams.

Since GitHub issues are mainly used to keep track of all issues and less for discussion, the response time is significantly higher than in the chat.[1] On average I got an answer to my comments on GitHub issues after 27 hours, though one of them has not been replied to so far.[19] I do not think the response time needs to be improved, but rather that old issues should be tidied up more frequently. I have noticed that some issues are already solved by other commits or became obsolete and have not been closed. A better system to process issues could be a big upgrade for the community. Similar holds true for pull requests. The core team is already discussing an improved review system for open PRs which includes different labels for the status like "review ready", "nearly complete", "has useful work" and "no progress" (Vishnu KS, personal communication, February 15, 2021).

Lastly I want to mention the incredibly useful check-in stream in Zulip's community chat. People, especially frequent contributors and core team members, post their current status and work capacity.[27] Tim Abbott often reports the number of his unread messages and which tasks he will focus on during the next days. This creates transparency and makes it much easier for contibutors to wait for his comments or code reviews.[23]

# 5 Inside Zulip

## 5.1 Open-source philosophy

Open-source is not only a term to denote practical implications on software, but also the values it is based on. Zulip's dedication to FLOSS and its values are the result of Tim Abbott's belief in them, since he was the one that open-sourced the project in 2015 and is still the lead developer.[1] Even back in the times when the software was still proprietary, he had aspirations of open-sourcing it, but the other founders did not agree with him.[25]

In the community chat, Abbott writes "I could probably write a 10-page essay on the reasons I really value building [Z]ulip as libre and open source software"([25], message

by Tim Abbott, April 11 2019, 10:21 PM). Creating the best software possible is an important goal for him, and he views open-source as an economically efficient way to achieve this. As an example, he mentions the decreased time for bug fixes compared to proprietary software. Open-source software gives him the freedom to reuse his previously written code or get inspired by it, unlike code written for proprietary software that is unavailable to him after leaving the company. He also agrees with Steve Howell, one of the original Zulip contributors, that writing code that can be read and learned from by everyone is very enriching. Tim Abbott himself has gained a lot of his technical expertise from open-source projects like Linux and GCC.[25]

## 5.2 Perspective of a core developer

After Zulip was open-sourced, Tim Abbott founded the company Kandra Labs whose main purpose is to maintain Zulip.[25] Currently there are seven full-time and two half-time employees (Tim Abbott, personal communication, March 11, 2021). They are usually responsible for specific parts of the software but also free to work on whatever they like to (Vishnu KS, personal communication, February 15, 2021). The developers are not tied to specific working hours and work almost fully remote, even though there is an office in San Francisco (Vishnu KS, personal communication within the guest lecture in the lab course[21], February 2, 2021). Their main communication tool is the Zulip chat and almost all the decision-making happens in public to maintain transparency for all contributors. Only topics like revenue, marketing and hiring are talked about in private (Vishnu KS, personal communication, February 15, 2021).

It is not clearly communicated to the community who is part of the core team. Even among themselves, not all employees know each other well. To address this, one of them, namely Mateusz Mandera, recently conducted an ask-me-anything in the chat.[20] Many contributors participated and asked questions about his work as well as personal topics. Since this was very well-received by the community, others will likely do more events like this one.

In a private conversation with Vishnu KS who works full-time for Kandra Labs, I found out that even for him, it can be challenging to find new issues to work on, which validates my own experience. In his opinion, this is because Zulip takes part in Google Summer of Code, so there are many people that try to make high-quality PRs to have a better chance of getting into the program. Consequently, most of the issues that are well-described and ready to be worked on get claimed quickly. As a reaction, he prefers

to work on parts of the project that he is very familiar with and knows off the top of his head what to work on next rather than being dependent on suitable GitHub issues (Vishnu KS, personal communication, February 15, 2021).

## 5.3  Tim Abbott's role

Tim Abbott, as the lead developer, has undeniably the most important role in the project. He makes the final review on all PRs and merges them. Only in very few cases do other members of the core team do this (Vishnu KS, personal communication, February 15, 2021).[20] Ownership of the Zulip community chat is limited to Tim Abbott plus a few administrators.

Whenever a project seems to be dependent on only one person, one might ask what would happen if this person leaves. As a maintainer of Void Linux, Leah Neukirchen witnessed the unforeseeable leave of the project's creator and shared her experience in her guest talk. This has been a rather unlucky case since he exclusively managed some of Void's central resources like the Void Linux GitHub Organisation, the IRC Channels, and the domains.[22] I do not assume something similar would happen to Zulip, and I am not even sure if its resources are only managed by Tim Abbott, but caution is still required.

Regardless of access rights to central resources, Zulip has proven over the last month that its community functions well even without its lead developer. Tim Abbott has been unable to work for most of November, December and January due to personal reasons.[23] Though this slowed down the rate of merging PRs by five to ten times,

**Aug 26, 2012 – Mar 4, 2021**                    Contributions: Commits ▾

Contributions to master, excluding merge commits



Figure 1: Contributions to Zulip's master branch, excluding merge commits, in the time period of August 26, 2012 to March 4, 2021 [24]

the project was able to run smoothly otherwise (Vishnu KS, personal communication, February 15, 2021). The decrease in PR merge rate can also be seen in Figure 1 which shows the merged commits. This is not too big of a deal, since even if the original author of a PR is not active anymore, other contributors will rebase it and will get it merged at some point. Usually PRs are only rejected if they do not address a current issue, or if another commit already address the issue (Vishnu KS, personal communication, February 15, 2021).

In my experience, Zulip benefits largely from Tim Abbott's decision power, as his veto power can resolve lengthy discussion, which makes Zulip's development move faster. A possible improvement could be to communicate a roadmap for the project to make it even clearer what the focus should be at the moment. Currently they only make use of the label "high priority" to indicate the most important issues (Vishnu KS, personal communication, February 15, 2021).

# 6 Zulip and Radically Open Security: parallels

Melanie Rieback is the founder of the non-profit computer security consultancy Radically Open Security. The company's goal is to make the world a better place rather than optimizing for profit. She sees her business as a prototype for an ideal company.[26] Comparing this to Zulip reveals major similarities. Tim Abbott's goal for Zulip is "to prove that one can build a great business that is entirely about making the world's best software, all open source"([25], message by Tim Abbott, April 11 2019, 10:21 PM). These are excellent examples of people who follow their values and aim to do good for society. Both of them do not see growth as a priority, but instead optimize on the positive impact of their work (Vishnu KS, personal communication within the guest lecture in the lab course[21], February 2, 2021; Melanie Rieback, personal communication within the guest lecture in the lab course[21], February 16, 2021).

One could ask why there are not more people building such great businesses. A likely answer would be lack of money. Getting a company through the first years without having investors is hard, and if growth is not one of the highest priorities, it is even harder. Both Abbott and Rieback were only able to follow their dreams because they had significant savings from their previous jobs and, in Abbott's case, from selling two start-ups. Even though they were both lucky enough to also get funding that was independent of the companies' development, their personal capital was crucial (Melanie Rieback, personal communication within the guest lecture in the lab course[21], February

16, 2021).[1] Seeing this side of the medal shows that it often is an extremely privileged position to be able to run a business like theirs. Unfortunately, it still seems essential to have been part of the growth-oriented business world to be financially independent and follow movements like post-growth entrepreneurship in our world today.

Resorting to funding through venture capital and investors is not a solution to this problem, since it would put constant pressure on the company to hit numbers and push the original mission into the background. According to Tim Abbott's experience, it is especially dangerous for open-source software. These businesses often end up changing their software to only open-core, because they are having a hard time making enough money to satisfy the investor. Zulip got investment offers from top firms, but Abbott refused all of them. He did not want to put the company at risk of losing its original values and purpose.[25] Radically Open Security, being a non-profit, is not even allowed to accept investor money, which is an even stronger safeguard for its values.

# 7 Remarks on preliminary report

After reading through my preliminary report again I want to add a few remarks that I came across during the last few months of engaging with Zulip.

- Dropbox's acquisition of Zulip in 2014, described in "1.2 Founding Zulip, Inc."[1], was mainly for technological reasons, since they wanted to build a chat software based on Zulip. This plan was dropped quickly because their main product still needed a lot of work. The secondary reason for the acquisition was Zulip's team which then started working for Dropbox.[25][28]

- Even though the main purpose of Kandra Labs remains Zulip's development as open-source project, like it is described in "1.4 Commercial (re-)launch"[1], mentoring programmers can be seen as its secondary mission.[25]

- Two members of the core team confirmed that usually only Tim Abbott merges PRs which was written in "3.1 Governance structure"[1] (Vishnu KS, personal communication, February 15, 2021).[20] Steve Howell as one of the longest-serving members of Zulip is an exception to this and has merged a significant amount of PRs over the last month.[29]

- My experience confirmes that Zulip optimized its contributon process to be easy and efficient which was mentioned in "3.1 Governance structure" and "5.1 Managing

contributions"[1]. I had a lot of fun engaging with the project and will likely continue to contribute to Zulip in the future. Building a great community has always been a goal for Tim Abbott. Back in 2017 25% of the GitHub issues were about improving the development experience.[30] Kandra Labs also hired a professional technical writer to help writing the documentation and a consultant for managing open-source projects in the early days.[28]

- Among the users listed in section "6 Users of the project"[1] our lab course can be added, since we recently moved from Rocket.Chat to Zulip.

# References

[1] Helena Klause, Preliminary report, `https://www21.in.tum.de/teaching/osp/WS20/assets/pr-klause-zulip.pdf`, last visited: 15.03.2021

[2] Discussion on improving the behavior of hotspots, `https://chat.zulip.org/#narrow/stream/9-issues/topic/disable.20helpful.20popup`, last visited: 15.03.2021

[3] Subsystem documentation for hotspots, `https://zulip.readthedocs.io/en/latest/subsystems/hotspots.html`, last visited: 15.03.2021

[4] PR for: improving the behavior of hotspots, `https://github.com/zulip/zulip/pull/16909`, last visited: 15.03.2021

[5] Issue for welcome emails, `https://github.com/zulip/zulip/issues/9220`, last visited: 15.03.2021

[6] Issue for custom emoji do not show up in search, `https://github.com/zulip/zulip/issues/17037`, last visited: 15.03.2021

[7] PR for typo in emoji documentation, `https://github.com/zulip/zulip/pull/17041`, last visited: 15.03.2021

[8] PR for welcoming emails, `https://github.com/zulip/zulip/pull/17081`, last visited: 15.03.2021

[9] Issue for invite system, `https://github.com/zulip/zulip/issues/6745`, last visited: 15.03.2021

[10] Discussion on invite system issue, `https://chat.zulip.org/#narrow/stream/49-development-help/topic/User.20Invite/near/976269`, last visited: 15.03.2021

[11] Issue for custom hotkeys, `https://github.com/zulip/zulip/issues/4777`, last visited: 15.03.2021

[12] Duplicate issue for custom hotkeys, `https://github.com/zulip/zulip/issues/11896`, last visited: 15.03.2021

[13] Issue for supporting hotkeys on international keyboard layouts, `https://github.com/zulip/zulip/issues/7517`, last visited: 15.03.2021

[14] Discussion on supporting hotkeys on international keyboard layouts, `https://chat.zulip.org/#narrow/stream/101-design/topic/custom.20hotkeys/near/1111446`, last visited: 15.03.2021

[15] PR for fixing a sentence in the documentation on caching, `https://github.com/zulip/zulip/pull/17207`, last visited: 15.03.2021

[16] Documentation for commit discipline, `https://zulip.readthedocs.io/en/latest/contributing/version-control.html#commit-discipline`, last visited: 15.03.2021

[17] Zulip's documentation, `https://zulip.readthedocs.io/en/latest/`, last visited: 15.03.2021

[18] All messages in the Zulip community chat sent by me, `https://chat.zulip.org/#narrow/sender/17784-user17784`, last visited: 15.03.2021

[19] All comments on Zulip GitHub issue by me, `https://github.com/zulip/zulip/issues?q=is%3Aissue+commenter%3Amrchntia`, last visited: 15.03.2021

[20] Mateusz Mandera's ask-me-anything, `https://chat.zulip.org/#narrow/stream/277-discussions/topic/AMA.20Mateusz`, last visited: 15.03.2021

[21] Practical course - contributing to an open-source project, `https://www21.in.tum.de/teaching/osp/WS20/index.html`, last visited: 15.03.2021

[22] Void Linux News in Mai 2018, `https://voidlinux.org/news/2018/05/serious-issues.html`, last visited: 15.03.2021

[23] Zulip's community chat stream "checkins" and topic "tabbott", `https://chat.zulip.org/#narrow/stream/65-checkins/topic/tabbott`, last visited: 15.03.2021

[24] History of merged commits, `https://github.com/zulip/zulip/graphs/contributors`, last visited: 15.03.2021

[25] Zulip's community chat stream "discussions" and topic "zulip history", `https://chat.zulip.org/#narrow/stream/277-discussions/topic/zulip.20history`, last visited: 15.03.2021

[26] Radically Open Security, `https://www.radicallyopensecurity.com/business-model.htm`, last visited: 15.03.2021

[27] Zulip's community chat stream "checkins", `https://chat.zulip.org/#narrow/stream/65-checkins`, last visited: 15.03.2021

[28] Zulip's community chat stream "discussions" and topic "Zulip OSS project history", `https://chat.zulip.org/#narrow/stream/277-discussions/topic/Zulip.20OSS.20project.20history`, last visited: 15.03.2021

[29] Merged PRs, `https://github.com/zulip/zulip/pulls?q=is%3Apr+is%3Amerged`, last visited: 15.03.2021

[30] Rishi Gupta, Growing an Open Source Community, `https://www.youtube.com/watch?v=lXFO2ULktEI`, last visited: 15.03.2021