

Gymnázium Thomase Manna

YEAREND WORK

Open Source

Filip Troníček

Thesis supervisor: **[REDACTED]**

Year: **2020/2021**

Grade: **5**

Table of Contents

1	Introduction	3
1.1	Goals	3
1.2	Methods of information gathering	3
2	What is open-source software (OSS)?	4
3	A brief history of open source	4
3.1	Open code	4
3.1.1	The decline of open code popularity in the 1970s	5
4	The open-source philosophy	6
5	Git hosting services	7
5.1	What even is git?	7
5.2	GitHub and GitLab	7
5.2.1	GitHub	7
5.2.2	GitLab	8
6	OSS funding and finances	8
6.1	Open Collective	8
6.2	GitHub Sponsors	9
7	Case studies	10
7.1	MediaWiki	10
7.2	The Linux kernel	10
7.3	Interclip	10
8	Conclusion	12
8.1	The conclusion of the conclusion	12
9	Sources	13
10	Attachments	14

1 Introduction

My seminar work is going to be all about open source. I want to discuss and explain most of the key terminology, history, and environment where open source finds itself in. This work is a lot about programming and computers, but also about people, our society, and awesome communities that were born thanks to OSS¹.

1.1 Goals

In addition to the goals I mentioned, I want to explain how I fit in the open-source sphere and how everyone can (and most likely should) contribute².

1.2 Methods of information gathering

The information gathered will most likely consist of two different media: the first being me, because I enjoy writing about open source, due to the fact I have been interested in the topic for a couple of years now. The other and probably more legitimate one is the internet, where I will do a lot of research, mainly because that is the place where open-source history lies.

¹ Open Source Software

² By coincidence, the phrase „everyone can contribute “, is one of the slogans of a big open-source company called [GitLab](#)

2 What is open-source software (OSS)?

“Open-source software is a type of computer software in which source code is released under a license in which the copyright holder grants users the rights to use, study, change, and distribute the software to anyone and for any purpose.”, (Wikipedia, 2021.)³

Let us dissect this to better understand what it means. It is clear that it is a type of software, that is the first part, and the second part says that the software is distributed under some terms of use (how the people interacting with the software should behave). These “terms” are by open-source projects very welcoming and open. One example would be the MIT license⁴ (which is amongst the most used OSS licenses out there), which states that anyone can copy the software and even sell it commercially, but they must attribute the original author of the software used. There are a lot of other open-source licenses, including but not limited to⁵: The Apache License 2.0, GPL⁶, Mozilla Public License 2.0.

3 A brief history of open source

The term “open source” was first coined in 1998 by an American forecaster called Christine Peterson, who worked at the time in the Foresight Institute, which first started using the term, but that is not actually where the history begins.

3.1 Open code

Source code was publicly shared for a very long time, in the 1950s companies who developed any kind of software shipped the source code of the software alongside the actual machine binaries, because they had to be equipped with the right drivers to work with the operating system of the customer(s). A lot of the drivers and modifications were openly shared by universities, the same way universities stick to keeping knowledge free and accessible. (Wikipedia, 2021)

³More reference about licensing code can be found in the book „Understanding Open Source and Free Software Licensing “, written by Andrew M. St. Laurent, which is available on Google Books: <https://books.google.com/books?id=04jG7TTLujoC&pg=PA4>

⁴How the MIT license looks like <https://mit-license.org/>

⁵A comparison of various OSS licenses: https://en.wikipedia.org/wiki/Comparison_of_free_and_open-source_software_licences

⁶GNU General Public License

3.1.1 The decline of open code popularity in the 1970s

After code development costs skyrocketed, companies started obfuscating their code, making it “close source”. Also, in 1974, source code was made subject to copyright under American law, which (alongside the decline of free and open code) prompted Richard Stallman to create a mechanism known as copyleft ⁷as a part of the Free Software Foundation. Copyleft was used for many of the early projects made by Stallman.

⁷More about copyleft: <https://en.wikipedia.org/wiki/Copyleft>

4 The open-source philosophy

There is for sure a lot to say about why people are doing open source, but a motivation for many is the act of forming communities and contributing to something bigger than themselves, as some projects can have a great impact on the lives of millions of people every day (projects like Wikimedia, Linux, Android OS, Chromium). Many others just want to share and show others what amazing projects they worked on and let others use the tools and apps they made. The third major motivation is profit; there is more money in OSS than you might think, which I further discuss in the “OSS funding and finances” chapter. It can be either from a job directly or through the means of sponsoring and other types of funding. Many companies (amongst which are most tech giants like Microsoft⁸, Google⁹, IBM¹⁰, or Apple^{11,12}) are releasing some projects as open source because it is both good PR for the company and advantageous for the project itself. After all, there is a lot of free labor to acquire from it.

⁸Microsoft’s Open-Source Blog: <https://cloudblogs.microsoft.com/opensource/>

⁹Google Open Source: <https://opensource.google/>

¹⁰IBM’s open-source homepage: <https://www.ibm.com/opensource/>

¹¹A list of open-source projects by Apple Inc.: <https://opensource.apple.com/>

¹²Apple’s organization on GitHub: <https://github.com/apple/>

5 Git hosting services

There are a lot of websites that provide the ability to host code for millions of users, some are free, and some are paid.

5.1 What even is git?

Git is a control-version system, which is in practice the backbone and the middleman between always writing new versions of your code and having a sustainable environment around it. Arguably the biggest advantage is the fact you can go infinitely back in the history of your code with every change being marked as a “commit”¹³. The second essential feature of git is branches. Branches are just copies of the same code, but ready to be worked on. Let us imagine a situation where you are developing a video game, and now you want to add a new level to it. You can easily start a new branch (for example *new-level*) and work on the feature. On your main branch¹⁴ you can still access the working code and provide your users with the most up-to-date working version and at the same time develop your new level. When you are finished, you can use a feature called branch merging, and just add your changes to your main branch. This is extremely handy for developers working on massive projects, where thousands of features are developed at the same time.

5.2 GitHub and GitLab

Both GitHub and GitLab are the biggest companies, which host code for free for individuals, making most of their money by selling on-premises or cloud solutions to enterprises.

5.2.1 GitHub

GitHub is the biggest code collaboration platform out there. Since 2008, they have been offering a tool to share code amongst developers. There is an ever-growing community of more than 56 million users (September 2020) along with tens of millions of code

¹³Git’s documentation about what a git commit is: <https://git-scm.com/docs/git-commit>

¹⁴Since the beginning of git, the default branch has been *master*, but it is currently transitioning to *main*, the code hosting platform GitHub has already made the change and now defaults all new projects to use the *main* branch: <https://www.theserverside.com/feature/Why-GitHub-renamed-its-master-branch-to-main>

repositories¹⁵. On October 26th, 2018, Microsoft acquired GitHub for \$7.5 million. GitHub hosts many of the world's biggest Open Source projects, including but not limited to [React.js](#), [TensorFlow](#), [Visual Studio Code](#), [Electron.js](#), [Windows Calculator](#), [Bitcoin's source code](#), [Python](#), and many others.

5.2.2 GitLab

GitLab is also a giant version control platform, that has about 30 million registered users¹⁶, making it GitHub's biggest competitor. What makes GitLab different is that most of its code is also Open Source (the biggest component which is not fully FLOSS¹⁷ is the enterprise edition of their software¹⁸), making it an obvious choice for many, alongside its many more features¹⁹. Even though it has many more features, it also lacks robustness and does not have that much brand appeal, as GitHub does.

6 OSS funding and finances

Money is an interesting aspect of open source, since if code is FLOSS, how can anyone make money off it? There are a couple of different approaches to financing OSS projects, but the most popular one is simply running the project out of donations made from your users as a sign of appreciation of your work. This is the case for many smaller open-source projects because, for bigger ones, individuals struggle to finance the projects themselves. For bigger projects, they have either corporate backing (e.g., Visual Studio Code is run by Microsoft), they somehow make money from their software (e.g., GitLab with their EE²⁰ licenses) or they have bigger sponsors like tech giants which invest tens of thousands of dollars in libraries and other projects they depend on.

¹⁵ „In software development, a repository is a central file storage location. It is used by version control systems to store multiple versions of files. While a repository can be configured on a local machine for a single user, it is often stored on a server, which can be accessed by multiple users. “, (Tech Terms, 2011).

¹⁶ As per their about page: <https://about.gitlab.com/company/>

¹⁷ FLOSS means Free, Libre, and Open-Source Software

¹⁸ Although the code itself is open source, the software cannot be used without a purchased license key, a blog post from GitLab explaining why: <https://about.gitlab.com/blog/2019/08/23/a-single-codebase-for-gitlab-community-and-enterprise-edition/>

¹⁹ GitLab's GitHub vs GitLab comparison page: <https://about.gitlab.com/devops-tools/github-vs-gitlab/>

²⁰ Enterprise edition

6.1 Open Collective

Open Collective is a company that helps open-source projects be sustainable, helping in fields like accounting, invoices, and donations. Another one of their main goals is to help companies support open-source projects and for the open-source projects to thrive from that support. You can also (as an open project) submit expenses and pay taxes so that everything under your non-profit remains legal. Many large projects use it because it provides a platform that is robust and works.

6.2 GitHub Sponsors

GitHub Sponsors is a new platform offered by GitHub since May 2019, which allows individuals and recently also organizations to be sponsored by anyone on their platform. GitHub covers the processing fees of the payments, which means 100% of the funds go directly to the developer(s). It is very simple to set up even for individuals, and it is an awesome way of appreciation of one's work, right there on GitHub.

7 Case studies

Because there are a lot of iconic OSS projects, I wanted to highlight at least a couple of them here.

7.1 MediaWiki

Since 2002, MediaWiki is the project that powers Wikipedia, the 13th most visited website on the internet²¹. It powers loads of other wikis as well, but Wikipedia is for sure the biggest one. It is a complete CMS²² for organizing media, pages and has a lot more features under the hood. Its source code is written in PHP and hosted on its self-hosted Gitiles solution, but it is moving to GitLab²³.

7.2 The Linux kernel²⁴

One of the world's most popular operating systems, GNU/Linux is powered by the Linux Kernel, which is fully open source. The Linux operating system is used in many places, although it is the least used Desktop OS. For example, most TVs, servers, Internet of Things devices, routers, and many other devices use Linux. Some computers require Linux because of its customizability (it can run on a little Raspberry Pi and can be used in a supercomputer at the same time). I skipped one huge product in which Linux is used, and that is the Android operating system, which accounts for about 86% of all phones on the globe (as of February 2020). (Craig Smith, 2020)

7.3 Interclip²⁵

If the last two projects rang a bell, this one for sure will not, and there is a good reason for it. The good reason I am talking about is the fact that it is me who made it. It is a simple open-source website for sharing URLs and files with your friends or other devices. I started this project way back in February 2019, with the project growing rapidly ever since. Thanks to its development, I have met many awesome developers and got a lot of neat feedback thanks to the app being open source and transparent about everything it does.

²¹ Stat from Wikipedia's Alexa ranking: <https://www.alexa.com/siteinfo/wikipedia.org>

²² Content Management System

²³ About MediaWiki moving to GitLab: https://www.mediawiki.org/wiki/GitLab_consultation

²⁴ Further reading: <https://www.kernel.org/category/about.html>, https://en.wikipedia.org/wiki/Linux_kernel.
Git repository: <https://git.kernel.org/pub/scm/linux/kernel/git/torvalds/linux.git>

²⁵ Interclip's website: <https://interclip.app/>, Interclip's GitHub repository: [interclip/interclip](https://github.com/interclip/interclip)

8 Conclusion

Open-source software is an awesome topic, which is a completely unknown term for many. This is a shame and hopefully, this document has helped you in understanding what and why it is.

8.1 The conclusion of the conclusion

In the goals, I said I want to discuss and describe the terms used in the open-source fields, which I did, including describing open-source software. I also shed light on a few cherry-picked OSS projects that impact our daily lives, including the fact that I do open-source work, too.

I did not meet all my goals though, because a lot of noteworthy projects just did not fit my length budget (examples include: the VLC player, GitLab, OBS, and Taskord).

9 Sources

Terms, T. (2011, August 18). Repository. Retrieved March 12, 2021, from <https://techterms.com/definition/repository>

Smith, C. (n.d.). 95 Amazing Android Statistics. Retrieved April 18, 2021, from <https://expandedramblings.com/index.php/android-statistics/>

Wikipedia contributors. (2021, January 25). Open-source software. In *Wikipedia, The Free Encyclopedia*. Retrieved 16:11, February 3, 2021, from https://en.wikipedia.org/w/index.php?title=Open-source_software&oldid=1002573677

Wikipedia contributors. (2021, March 29). History of free and open-source software. In *Wikipedia, The Free Encyclopedia*. Retrieved 18:23, April 18, 2021, from https://en.wikipedia.org/w/index.php?title=History_of_free_and_open-source_software&oldid=1014962889

10 Attachments

Lots of blank space, all the attachments are in the footnotes, so check them out if you missed it, otherwise, that is it!