Setting Guidelines

Yegor Bugayenko

Lecture #5 out of 8
80 minutes

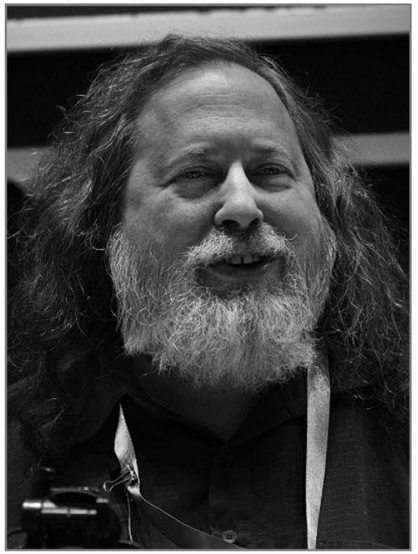
The slidedeck was presented by the author in this YouTube Video

1. Have `LICENSE.txt` file with an MIT license in your repository

Arnoud Engelfriet

"Perhaps the most difficult issue when setting up the project is which license to choose... No one will contribute code just because it's GPL or BSD. But with the wrong license, your chances of a successful open source release are slim."

— Arnoud Engelfriet. Choosing an Open Source License. *IEEE Software*, 27(1): 48–49, 2009. doi:10.1109/MS.2010.5



Richard Stallman

"**copyleft**: Everyone will be permitted to modify and redistribute GNU, but no distributor will be allowed to restrict its further redistribution. That is to say, proprietary modifications will not be allowed. I want to make sure that all versions of GNU remain free."

— Richard Stallman. The GNU Manifesto, 1985

Which license to choose?

License	Type
Massachusetts Institute of Technology (**MIT**)	Copyright
Apache	Copyright
Berkeley Software Distribution (**BSD**)	Copyright
GNU Public License (**GPL**)	Copyleft

What permissions may be granted by a license?

- Permission to use
- Permission to modify
- Permission to distribute
- Permission to not mention the product
- Permission to remove copyright notice

Source: Yi-Hsuan Lin, Tung-Mei Ko, Tyng-Ruey Chuang, Kwei-Jay Lin, et al. Open Source Licenses and the Creative Commons Framework: License Selection and Comparison, 2006

Table 1. Ranking of FOSS licenses' degree of *openness* based on eight specific considerations.

(+): higher degree of *openness*; (-) lower degree of *openness*

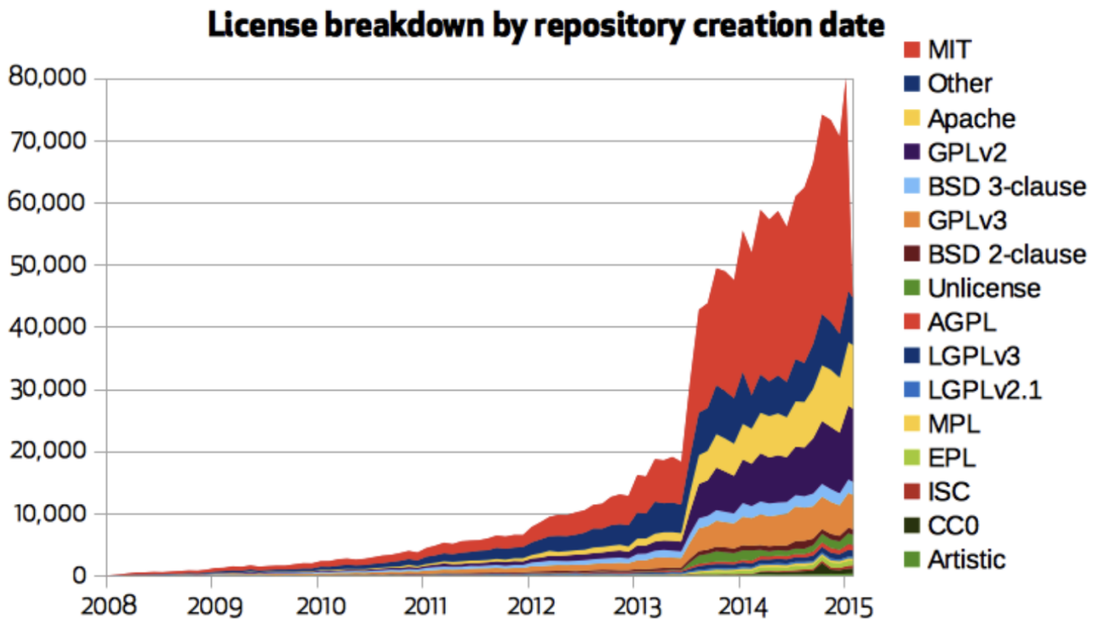
License type	Licenses	(1) Collection of loyalties is allowed when a program is distributed	(2) The source code is provided when the original program is redistributed	(3) The source code must be provided when a modification is distributed	(4) A modification should be distributed under the same license as the original program	(5) A fee higher than the distribution cost can be collected when a program is distributed without the source code	(6) The program can be sublicensed	(7) A documentation must be provided with a documentation	(8) A fee higher than the distribution cost can be collected when a program is distributed with the source code	Degree of *openness*
GPL type	GPL	No(+)	Yes(+)	Yes(+)	Yes(+)	No(+)	No(+)	Yes(+)	Yes(-)	1
	LGPL	No(+)	Yes(+)	Yes(+)	Yes(+)	No(+)	No(+)	Yes(+)	Yes(-)	1
Others	MPL	No(+)	Yes(+)	Yes(+)	Yes(+)	No(+) (source code is always redistributed)	Yes(-)	Yes(+)	Yes(-)	2
	QPL	No(+)	Yes(+)	Yes(+)	No(-)	No(+)	No(+)	No(-)	Yes(-)	3
	CPL	No(+)	Yes(+)	Yes(+)	No(-)	No(+)	Yes(-)	No(-)	Yes(-)	4
	Artistic	No(+)	Yes(+)	No(-)	No(-)	No(+) (source code is always Redistributed)	No(+)	Yes(+)	No(+)	5
BSD type	Apache v.2.0	Yes(-)	No(-)	No(-)	No(-)	Yes(-)	No(+)	Yes(+)	Yes(-)	6
	Zlib	Yes(-)	No(-)	No(-)	No(-)	Yes(-)	No(+)	Yes(+)	Yes(-)	6
	Apache v.1.1	Yes(-)	No(-)	No(-)	No(-)	Yes(-)	No(+)	No(-)	Yes(-)	7
	BSD	Yes(-)	No(-)	No(-)	No(-)	Yes(-)	No(+)	No(-)	Yes(-)	7
	MIT	Yes(-)	No(-)	No(-)	No(-)	Yes(-)	Yes(-)	No(-)	Yes(-)	8

Source: Yi-Hsuan Lin, Tung-Mei Ko, Tyng-Ruey Chuang, Kwei-Jay Lin, et al. Open Source Licenses and the Creative Commons Framework: License Selection and Comparison, 2006

Ben Balter

"Unsurprisingly, MIT, Apache, and GPL are the clear front runners, with some 15% of licensed projects opting for a non-standard license."

— Ben Balter. Open Source License Usage on GitHub.com. https://jttu.net/github2015licenses, 3 2015. [Online; accessed 20-03-2024]



Source: Ben Balter. Open Source License Usage on GitHub.com. https://jttu.net/github2015licenses, 3 2015. [Online; accessed 20-03-2024]

2. Make the `README.md` file attractive [Bugayenko, 2019].

Mandatory sections of `README.md`:

- Title, logo, badges
- What is it? What problem does it solve?
- How to quick start?
- How to contribute?
- Who is who?

Source: Yegor Bugayenko. Elegant READMEs. `https://www.yegor256.com/190423.html`, 4 2019. [Online; accessed 05-03-2024]

Gede Artha Azriadi Prana

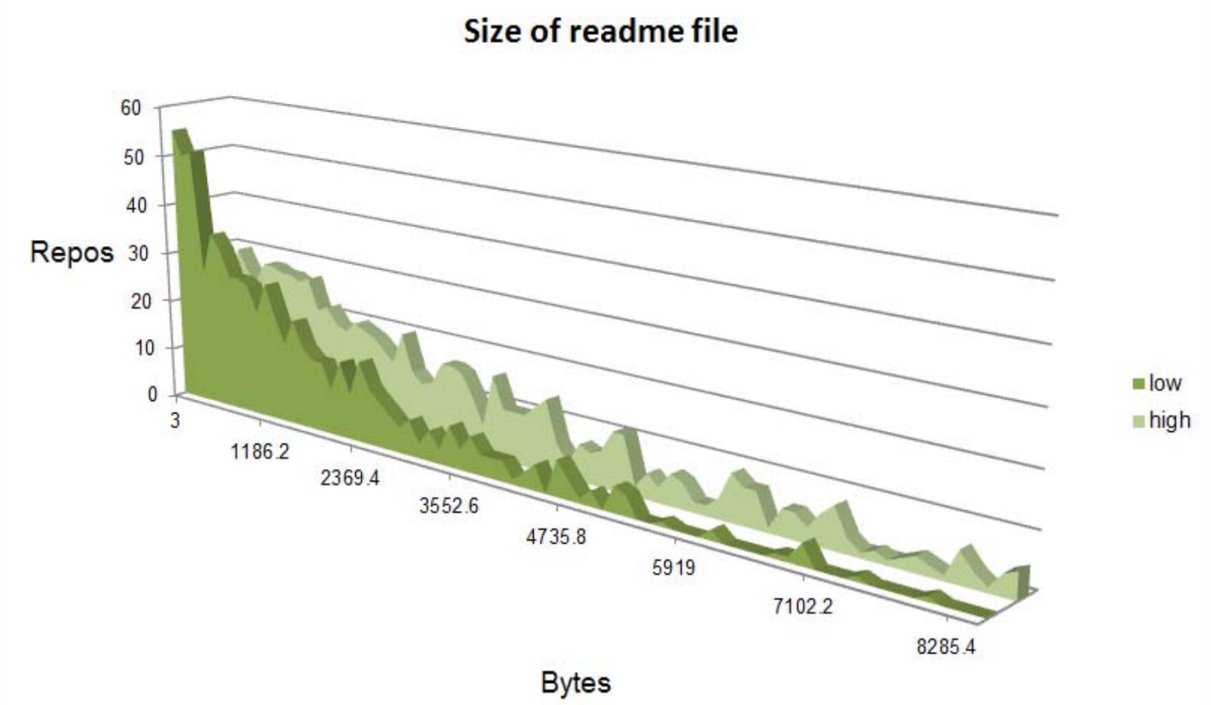
"We conduct a qualitative study involving the manual annotation of 4,226 README file sections from 393 randomly sampled GitHub repositories. We find that information discussing the 'What' and 'How' of a repository is very common, while many README files lack information regarding the purpose and status of a repository."

— Gede Artha Azriadi Prana, Christoph Treude, Ferdian Thung, Thushari Atapattu, and David Lo. Categorizing the Content of GitHub README Files. *Empirical Software Engineering*, 24(1):1296–1327, 2019. doi:10.1007/s10664-018-9660-3

Simon Weber

"Upon investigation, popular projects were found to have larger READMEs (median 2 kilobytes vs. 500 bytes). Also, 95% of popular projects have nonempty READMEs, compared to only 65% of unpopular projects."

— Simon Weber and Jiebo Luo. What Makes an Open Source Code Popular on GitHub? In *Proceedings of the International Conference on Data Mining Workshop*, pages 851–855. IEEE, 2014. doi:10.1109/ICDMW.2014.55



Source: Simon Weber and Jiebo Luo. What Makes an Open Source Code Popular on GitHub? In *Proceedings of the International Conference on Data Mining Workshop*, pages 851–855. IEEE, 2014. doi:10.1109/ICDMW.2014.55

Asher Trockman

"We find that non-trivial badges, which display the build status, test coverage, and up-to-dateness of dependencies, are mostly reliable signals, correlating with more tests, better pull requests, and fresher dependencies."

— Asher Trockman, Shurui Zhou, Christian Kästner, and Bogdan Vasilescu. Adding Sparkle to Social Coding: An Empirical Study of Repository Badges in the *npm* Ecosystem. In *Proceedings of the 40th International Conference on Software Engineering*, pages 511–522, 2018. doi:10.1145/3180155.3180209

Shaowei Wang

"The frequency/number of readme updates and the number of lists and links positively correlate with the likelihood of a repository being popular."

— Tianlei Wang, Shaowei Wang, and Tse-Hsun Peter Chen. Study the Correlation Between the Readme File of GitHub Projects and Their Popularity. *Journal of Systems and Software*, 205(1), 2023. doi:10.1016/j.jss.2023.111806

> 3. Make the `master` branch read-only [Bugayenko, 2014].

4. Put `CODE_OF_CONDUCT.md` file to your repository... not

Parastou Tourani

"We found that the top codes of conduct are adopted by hundreds to thousands of projects, while all of them share five common dimensions."

— Parastou Tourani, Bram Adams, and Alexander Serebrenik. Code of Conduct in Open Source Projects. In *Proceedings of the 24th International Conference on Software Analysis, Evolution and Reengineering (SANER)*, pages 24–33. IEEE, 2017. doi:10.1109/SANER.2017.7884606

TABLE I: ORDER OF MAGNITUDE FOR CODES OF CONDUCT OBTAINED FROM GITHUB (FIRST APPROACH).

Code of Conduct	Number of Hits on GitHub	Examples
Contributor Covenant	43,681	Molajo/Molajo, trevororeilly/dotfiles, SecComm/Crayon, yuluyi/Isomorphic-React-Seed, tweetstockr
Open Code of Conduct	2,167	wildland/cli-tools, KineticCafe/code-of-conduct, Tacklr/CacheManager, spotify/ios-ci, PearlCast/PearlCast
Python	2,025	PyDiff/PyDiff.github.io, 18F, brettcannon/oplop, link39/205-pi, roadcap/homebrew,sfdevs/sdcodecamp
Citizen	1,253	npr/npr-one-api-js-sdk, cworth-gh/stony, gulpjs/gulp, lkodai/Design-LK, ojs/ojs, ctdk/goiardi
Ubuntu	1,180	goodeggs/format-location, Star2Billing/cdr-stats-docs, garyjs/Newfiesautodialer,Alamofire/Foundation, Trustroots/trustroots
Django	1,054	jrief/django-angular, DBCboots, Pythonkc, Calagator, ordergroove/check_mariadb_slaves
Geek Feminism	544	nzruby, brave/chromium, crosswalk-project/chromium-crosswalk, javascripthers/javascripthers.github.io, openSNP/snpr

Source: Parastou Tourani, Bram Adams, and Alexander Serebrenik. Code of Conduct in Open Source Projects. In *Proceedings of the 24th International Conference on Software Analysis, Evolution and Reengineering (SANER)*, pages 24–33. IEEE, 2017. doi:10.1109/SANER.2017.7884606

of their major components and ingredients. In particular, each code of conduct included the following five components in one form or the other:

- Purpose: the rationale for the code of conduct, typically the desire to obtain a certain kind of environment for project members to work and collaborate in.
- Honorable behaviour: behaviour that is valuable for and accepted by the community.
- Unacceptable Behaviour: negative behaviour that should be avoided.
- Enforcement: mechanisms for reporting and punishing violations of the code of conduct.
- Scope: the online and offline spaces where the code of conduct applies, for example only in the mailing list versus in any online discussion forum.

Source: Parastou Tourani, Bram Adams, and Alexander Serebrenik. Code of Conduct in Open Source Projects. In *Proceedings of the 24th International Conference on Software Analysis, Evolution and Reengineering (SANER)*, pages 24–33. IEEE, 2017. doi:10.1109/SANER.2017.7884606

"The Contributor Covenant explicitly mentions sexualized language or imagery, trolling, insulting and publishing of private information of others as unexpected behaviors. Django adds discriminatory jokes and violent threats to this list. Geek Feminism and the Open code of conduct provide a more detailed list, ..."

Django Code of Conduct

“We strive to be a community that welcomes and supports people of all backgrounds and identities. This includes, but is not limited to members of any race, ethnicity, culture, national origin, colour, immigration status, social and economic class, educational level, sex, sexual orientation, gender identity and expression, age, size, family status, political belief, religion, and mental and physical ability.”

Hana Frluckaj

"Even though including a code of conduct is considered a best practice recommendation to attract newcomers, there is, as of yet, no empirical evidence backing up that claim."

— Renee Li, Pavitthra Pandurangan, Hana Frluckaj, and Laura Dabbish. Code of Conduct Conversations in Open Source Software Projects on Github. *Proceedings of the ACM on Human-Computer Interaction*, 5(1):1–31, 2021. doi:10.1145/3449093

Dabbish Laura

"We found that projects in our sample did not extensively discuss the addition or changes to the code of conduct."

— Renee Li, Pavitthra Pandurangan, Hana Frluckaj, and Laura Dabbish. Code of Conduct Conversations in Open Source Software Projects on Github. *Proceedings of the ACM on Human-Computer Interaction*, 5(1):1–31, 2021. doi:10.1145/3449093

Jack Jamieson

"We found multiple significant relationships between value-related discussions and turnover, including that discussions about respectfulness predict an increase in contributors leaving and a decrease in new contributors, while discussions about social power predicted better contributor retention."

— Jack Jamieson, Naomi Yamashita, and Eureka Foong. Predicting Open Source Contributor Turnover From Value-Related Discussions: An Analysis of GitHub Issues. In *Proceedings of the 46th International Conference on Software Engineering*, pages 667–679. IEEE, 2023. doi:10.1145/3597503.3623340

5. Don’t explain coding guidelines, setup GitHub Action checks

Timothy Kinsman

"We analyzed the effect of adoption across 926 projects that had adopted GitHub Actions for at least 6 months. Our findings indicate that, on average, there are more rejected pull requests and fewer commits on merged pull requests after adopting GitHub Actions."

— Timothy Kinsman, Mairieli Wessel, Marco A. Gerosa, and Christoph Treude. How Do Software Developers Use GitHub Actions to Automate Their Workflows? In *Proceedings of the 18th International Conference on Mining Software Repositories (MSR)*, pages 420–431. IEEE, 2021. doi:10.1109/MSR52588.2021.00054

Alexandre Decan

"We observed that workflows tend to be used in the more active GitHub projects (more contributors, pull requests, commits and issues)."

— Alexandre Decan, Tom Mens, Pooya Rostami Mazrae, and Mehdi Golzadeh. On the Use of GitHub Actions in Software Development Repositories. In *Proceedings of the International Conference on Software Maintenance and Evolution (ICSME)*, pages 235–245. IEEE, 2022. doi:10.1109/ICSME55016.2022.00029

TABLE I
NUMBER AND PROPORTION OF REPOSITORIES USING GHA WORKFLOWS, GROUPED BY MAIN PROGRAMMING LANGUAGE.

	repositories		using GHA workflows	
language	**#**	**%**	**% language**	**% repo.**
JavaScript	13,542	19.6%	34.9%	15.9%
Python	12,319	17.8%	45.9%	19.0%
TypeScript	6,362	9.2%	58.5%	12.5%
Java	6,105	8.8%	39.2%	8.0%
C++	5,701	8.2%	40.9%	7.8%
Go	4,988	7.2%	57.2%	9.6%
C	4,314	6.2%	36.1%	5.2%
PHP	4,005	5.8%	48.2%	6.5%
C#	3,630	5.3%	34.6%	4.2%
Ruby	2,599	3.8%	50.8%	4.4%
Shell	2,327	3.4%	33.2%	2.6%
Swift	1,411	2.4%	34.4%	1.6%
Kotlin	1,150	1.7%	56.9%	2.2%
other	*694*	*1.0%*	*17.7%*	*0.4%*

Source: Alexandre Decan, Tom Mens, Pooya Rostami Mazrae, and Mehdi Golzadeh. On the Use of GitHub Actions in Software Development Repositories. In *Proceedings of the International Conference on Software Maintenance and Evolution (ICSME)*, pages 235–245. IEEE, 2022. doi:10.1109/ICSME55016.2022.00029

TABLE II
COMPARISON OF CHARACTERISTICS FOR GITHUB REPOSITORIES WITH AND WITHOUT GHA WORKFLOWS.

	median		effect size	
characteristic	**with**	**without**	**Cliff's δ**	**interpretation**
pull requests	124	41	0.384	*medium*
contributors	20	11	0.277	*small*
commits	598	344	0.229	*small*
issues	105	59	0.227	*small*
branches	5	4	0.139	*negligible*
age (months)	71	77	−0.082	*negligible*
stars	398	334	0.078	*negligible*
size (MB)	5,878	5,099	0.025	*negligible*
forks	84	80	0.018	*negligible*
watchers	24	25	−0.013	*negligible*

Source: Alexandre Decan, Tom Mens, Pooya Rostami Mazrae, and Mehdi Golzadeh. On the Use of GitHub Actions in Software Development Repositories. In *Proceedings of the International Conference on Software Maintenance and Evolution (ICSME)*, pages 235–245. IEEE, 2022. doi:10.1109/ICSME55016.2022.00029

Check this repo for inspiration: sdras/awesome-actions (a curated list of GitHub Action plugins for different purposes).

Some exotic GitHub Action plugins:

- hadolint-action for `Dockerfile`
- markdownlint-action for `README.md`
- shellcheck-action for Bash `.sh` scripts
- checkmake-action for `Makefile`
- action-yamllint for `YAML` files
- bibcop-action for BibTeX `.bib` files

Bibliography

Ben Balter. Open Source License Usage on GitHub.com. `https://jttu.net/github2015licenses`, 3 2015. [Online; accessed 20-03-2024].

Yegor Bugayenko. Master Branch Must Be Read-Only. `https://www.yegor256.com/140721.html`, 7 2014. [Online; accessed 05-03-2024].

Yegor Bugayenko. Elegant READMEs. `https://www.yegor256.com/190423.html`, 4 2019. [Online; accessed 05-03-2024].

Alexandre Decan, Tom Mens, Pooya Rostami Mazrae, and Mehdi Golzadeh. On the Use of GitHub Actions in Software Development Repositories. In *Proceedings of the International Conference on Software Maintenance and Evolution (ICSME)*, pages 235–245. IEEE, 2022. doi:10.1109/ICSME55016.2022.00029.

Arnoud Engelfriet. Choosing an Open Source License. *IEEE Software*, 27(1):48–49, 2009. doi:10.1109/MS.2010.5.

Jack Jamieson, Naomi Yamashita, and Eureka Foong. Predicting Open Source Contributor Turnover From Value-Related Discussions: An Analysis of GitHub Issues. In *Proceedings of the 46th International Conference on Software Engineering*, pages 667–679. IEEE, 2023. doi:10.1145/3597503.3623340.

Timothy Kinsman, Mairieli Wessel, Marco A. Gerosa, and Christoph Treude. How Do Software Developers Use GitHub Actions to Automate Their Workflows? In *Proceedings of the 18th International Conference on Mining Software Repositories (MSR)*, pages 420–431. IEEE, 2021. doi:10.1109/MSR52588.2021.00054.

Renee Li, Pavitthra Pandurangan, Hana Frluckaj, and Laura Dabbish. Code of Conduct Conversations in Open Source Software Projects on Github. *Proceedings of the ACM on Human-Computer Interaction*, 5(1):1–31, 2021. doi:10.1145/3449093.

Yi-Hsuan Lin, Tung-Mei Ko, Tyng-Ruey Chuang, Kwei-Jay Lin, et al. Open Source Licenses and the Creative Commons Framework: License Selection and Comparison, 2006.

Gede Artha Azriadi Prana, Christoph Treude, Ferdian Thung, Thushari Atapattu, and David Lo. Categorizing the Content of GitHub README Files. *Empirical Software Engineering*, 24(1):1296–1327, 2019. doi:10.1007/s10664-018-9660-3.

Richard Stallman. The GNU Manifesto, 1985.

Parastou Tourani, Bram Adams, and Alexander Serebrenik. Code of Conduct in Open Source Projects. In *Proceedings of the 24th International Conference on Software Analysis, Evolution and Reengineering (SANER)*, pages 24–33. IEEE, 2017. doi:10.1109/SANER.2017.7884606.

Asher Trockman, Shurui Zhou, Christian Kästner, and Bogdan Vasilescu. Adding Sparkle to Social Coding: An Empirical Study of Repository Badges in the *npm* Ecosystem. In *Proceedings of the 40th International Conference on Software Engineering*, pages 511–522, 2018. doi:10.1145/3180155.3180209.

Tianlei Wang, Shaowei Wang, and Tse-Hsun Peter Chen. Study the Correlation Between the Readme File of GitHub Projects and Their Popularity. *Journal of Systems and Software*, 205(1), 2023. doi:10.1016/j.jss.2023.111806.

Simon Weber and Jiebo Luo. What Makes an Open Source Code Popular on GitHub? In *Proceedings of the International Conference on Data Mining Workshop*, pages 851–855. IEEE, 2014. doi:10.1109/ICDMW.2014.55.