

# FLOSS Best Practices for Bots & Tools

A project that is valuable for an on-wiki workflow should protect the community by following best practices. Developers put in a lot of effort to build new things and keep them running. No one should feel that they must be available 24/7/365 to support their project. Popular projects will experience issues at all times of the day and night. By adopting a few simple practices, a tool maintainer can make it easier for others to help them keep their tool running.



## Pick a license

A software license explains the rights that you grant to others who wish to use or modify your software. All machine-readable software is implicitly granted copyright as a 'literary work'. Without an explicit license, the author retains all rights and prevents reproduction, distribution, or creation of derivative works until 70 years after the author's death. This is counter to the general principles of the Wikimedia movement and is a violation of the Toolforge terms of use.

Need help choosing a license? Visit <https://choosealicense.com/> for a guide to various FLOSS licenses.



## Publish the code

The Wikimedia movement promotes open knowledge. Publishing your source code has numerous potential benefits and few downsides. It is easy to setup a git repository providing public browsing and downloads. The Wikimedia Foundation, BitBucket, GitHub, and others provide this as a gratis service. Wikimedia Foundation hosting is also libre if that is important to you. As a bonus, you get a backup of your code.



## Have co-maintainers

Everybody needs a day off now and then. Sometimes we even need a long wiki break to recharge. Having multiple maintainers makes taking time off easier without forcing your users or the Toolforge admins deal problems during your absence. A co-maintainer doesn't have to be a highly technical person. At the bare minimum, they should be able to restart things when they are down and shout for help when more complex problems occur.

Don't forget to update your tool's documentation so that others know who to poke when they have issues!



## Write some docs

Write enough documentation so that every maintainer can help with restarts and basic troubleshooting. Where? A wiki page is always ready for you in the Tool namespace on Wikitech, but anywhere is fine. Add a MAINTAINERS file to your git repo, make a subpage in your User namespace on Wikitech or your home wiki; put it wherever you like, but put it somewhere.



## Going beyond

Use a public bug tracker, and check it for bug reports. • Put your maintenance documentation on Wikitech in the Tool namespace so it's easy to find. • Put end-user documentation on your favorite wiki, and link to it from your tool itself and/or the Wikitech page for your tool. • Build your tool using other Wikimedia and 3rd-party FLOSS projects (API client libraries, application frameworks) to save time and to make it easier for others to contribute. • Make a new tool for each unrelated activity. Attracting co-maintainers is easier when there is one thing that the tool does well.