

Next steps

2 minutes

You've added context to an issue, contributed a code review, and maybe even submitted a pull request of your own. Now, you want to immerse yourself further in the community around the project.

Get involved in the community

You'll find frequent contributors to the project in the comment section for issues and pull requests. Or, you can select **Insights** in the repository's navigation, and then select **Contributors** to find other active community members. Visit their GitHub profiles. Sometimes they'll suggest ways to get in touch with them.

You can also [follow organizations and enterprises on GitHub](#) to stay in touch. Your personal dashboard shows public activity for every enterprise, user, or organization you follow.

You can also find like-minded folks by attending meetups or conferences on open-source topics. Or, you can meet people if the project or ecosystem is large enough around the project you're interested in. Find archives with talk recordings for past events, podcasts, newsletters, and mailing lists.


Some projects have centralized communication, which is often referenced on the project's website or in the README file. There might be a Discord server, a Slack community, Gitter, IRC, or even regular "office hours."

Code reusability

Code, and solutions, can sometimes be reused across projects. Have you solved a very scoped issue for one project? Chances are other projects can benefit from it as well. You can:

- Publish as a stand-alone library (dependency).
- Mirror the project with your added functionality.
- [Create a GitHub Action](#) for others to include in their workflow.

The first option is probably the best course of action when your bit of code is like a plug-in that could be used across web-development projects. Mirroring or forking a project with the addition of your code is useful when you're solving a narrow use case for a small subset of customers, or even a single customer. Consider that you'll need to keep your fork up to date with the upstream repository if you want to benefit from (for instance) security patches.

GitHub Actions are packaged scripts that automate tasks in a software-development workflow in GitHub. The two different types of actions are container actions and JavaScript actions. You can submit your action to the GitHub Marketplace for discoverability. [GitHub Marketplace](#)  connects you to developers who want to extend and improve their GitHub workflows. Use this platform to publish actions and share apps with other users for free.

For all three of the suggested paths, consider that you're now a maintainer of a project. People will come to you with praise, questions, and complaints. Are you ready for such a commitment?

If your project takes off, people's apps might depend on your bit of code. Can you involve more people to take some of the potential load off? Do you have time to add documentation, triage issues, and review suggestions from people you've likely never met before? Consider your "bandwidth," and instead set expectations in your project's README file. Or, you can release your code in a public gist or a blog post. Code doesn't need to be on GitHub to be open source, after all.

Next unit: Knowledge check

[Continue >](#)