

# Contribute to the Docs

The LayerZero documentation uses a docs-as-code approach, meaning documentation is created using the same tools as code. The contribution workflow involves proposing changes to the docs by creating forks and pull requests (PRs) on the documentation GitHub repositories. This facilitates open contributions, testing, and review.

## Steps

To contribute changes:

1. Choose the repository you'd like to contribute to. See the list of LayerZero documentation repositories.

note These steps only apply to the doc sites that use Docusaurus. 1. In the repository, search for an existing issue to work on, or create a new issue describing the documentation issue you'd like to address. Make sure no one else is assigned to the issue, and assign yourself to it. If you don't have permission to assign yourself to it, leave a comment on the issue or contact a maintainer of that repository. 2. Fork the repository to your personal account. 3. Clone your forked repository to your computer.

git clone 1. Add an upstream remote.

git remote add upstream 1. Create and checkout a topic branch, naming it appropriately. We recommend including the issue number and a short description in the branch name (for example, 183-doc-cli-option 2. ), which is a reminder to fix only one issue in a PR.

git checkout -b - tip You can use a Git client such as Fork instead of the command line. 1. Open the repository in a text editor of your choice (for example, VS Code) and make your documentation changes. Make sure to follow the style guidelines and format your Markdown correctly.

info If you delete, rename, or move a documentation file, make sure to add a redirect. 1. Preview your changes locally to check that the changes render correctly. 2. Add and commit your changes, briefly describing your changes in the commit message. Push your changes to the remote origin.

git add \* git commit -m "" git push origin 1. On the original repository on GitHub, you'll see a banner prompting you to create a PR with your recent changes. Create a PR, describing your changes in detail. Link the issue that your PR fixes by adding fixes# 2. to the PR description. 3. If your PR fails any checks, displayed at the bottom of the PR page, fix those errors. If you want to include a new word that causes a spell check error, you can add that word to the project-words.txt file. 4. For most doc repositories, specific reviewers are automatically requested when you submit a PR. You can request additional reviewers in the right sidebar of your PR – for example, the original issue raiser. Make any required changes to your PR based on reviewer feedback, repeating steps 7–9. 5. After your PR is approved by two reviewers, all checks have passed, and your branch has no conflicts with the main branch, you can merge your PR. If you don't have merge access, a maintainer will merge your PR for you. You can delete the topic branch after your PR is merged.

Thank you for contributing to the docs! [Edit this page](#)

[Previous Support](#) [Next Hackathon Success](#)