# Contributing

Contributions are \*\*welcome\*\* and will be fully \*\*credited\*\*.

Please read and understand the contribution guide before creating an issue or pull request.

## Etiquette

This project is open source, and as such, the maintainers give their free time to build and maintain the source code

held within. They make the code freely available in the hope that it will be of use to other developers. It would be

extremely unfair for them to suffer abuse or anger for their hard work.

Please be considerate towards maintainers when raising issues or presenting pull requests. Let's show the

world that developers are civilized and selfless people.

It's the duty of the maintainer to ensure that all submissions to the project are of sufficient

quality to benefit the project. Many developers have different skillsets, strengths, and weaknesses. Respect the maintainer's decision, and do not be upset or abusive if your submission is not used.

## Viability

When requesting or submitting new features, first consider whether it might be useful to others. Open

source projects are used by many developers, who may have entirely different needs to your own. Think about

whether or not your feature is likely to be used by other users of the project.

## Procedure

Before filing an issue:

- Attempt to replicate the problem, to ensure that it wasn't a coincidental incident.

- Check to make sure your feature suggestion isn't already present within the project.

- Check the pull requests tab to ensure that the bug doesn't have a fix in progress.

- Check the pull requests tab to ensure that the feature isn't already in progress.

Before submitting a pull request:

- Check the codebase to ensure that your feature doesn't already exist.

- Check the pull requests to ensure that another person hasn't already submitted the feature or fix.

## Requirements

If the project maintainer has any additional requirements, you will find them listed here.

- \*\*[PSR-2 Coding Standard](https://github.com/php-fig/fig-standards/blob/master/accepted/PSR-2-coding-style-guide.md)\*\* - The easiest way to apply the conventions is to install [PHP Code Sniffer](http://pear.php.net/package/PHP\_CodeSniffer).

- \*\*Add tests!\*\* - Your patch won't be accepted if it doesn't have tests.

- \*\*Document any change in behaviour\*\* - Make sure the `README.md` and any other relevant documentation are kept up-to-date.

- \*\*Consider our release cycle\*\* - We try to follow [SemVer v2.0.0](http://semver.org/). Randomly breaking public APIs is not an option.

- \*\*One pull request per feature\*\* - If you want to do more than one thing, send multiple pull requests.

- \*\*Send coherent history\*\* - Make sure each individual commit in your pull request is meaningful. If you had to make multiple intermediate commits while developing, please [squash them](http://www.git-scm.com/book/en/v2/Git-Tools-Rewriting-History#Changing-Multiple-Commit-Messages) before submitting.

\*\*Happy coding\*\*!