# Contributing to Prism

Welcome! We would love to have you contribute bug fixes or new functionality to Prism.

The best starting point is to enter an \_\_issue\_\_ [here](https://github.com/PrismLibrary/Prism/issues). We can then have a brief discussion on what you want to do and where it fits with our milestones and goals for the library. As long as it sounds like something we would want to add to Prism, we will give you a thumbs up and ask for a pull request.

When you submit a \_\_pull request\_\_, there are a few things we would like you to comply with:

- New functionality must have accompanying unit tests with "good" code coverage if it is logic code that can be unit tested (i.e. not view stuff touching UI or platform APIs)

- Changes to existing functionality needs to be checked that it does not break any existing unit tests. If it does, then fixes to the unit test may be appropriate, but only if those changes maintain the original intent of the test.

- Some basic coding standard guidelines to start with:

- no leading "this."

- single type per file

- interface-based design to preserve testability and extensibility

- due consideration for inheritance (i.e. consider carefully whether something should be protected or virtual)

- member variables have leading underscore and are \_camelCased

- local variables are camelCased with no prefix

- types, properties, methods, events are PascalCased

- use new C# features (e.g nameof) where possible, but keep the code readable (e.g. don't var everything)

## Contribution License Agreement

You must sign a [.NET Foundation Contribution License Agreement (CLA)](http://cla.dotnetfoundation.org) before your PR will be merged. This a one-time requirement for projects in the .NET Foundation. You can read more about [Contribution License Agreements (CLA)](http://en.wikipedia.org/wiki/Contributor\_License\_Agreement) on wikipedia.

However, you don't have to do this up-front. You can simply clone, fork, and submit your pull-request as usual.

When your pull-request is created, it is classified by a CLA bot. If the change is trivial, i.e. you just fixed a typo, then the PR is labelled with `cla-not-required`. Otherwise it's classified as `cla-required`. In that case, the system will also tell you how you can sign the CLA. Once you signed a CLA, the current pull-request will be labeled `cla-signed` and all future pull-requests as `cla-already-signed`.