# Contributing to CodeIgniter4

## Contributions

We expect all contributions to conform to our [style guide](https://github.com/codeigniter4/CodeIgniter4/blob/develop/contributing/styleguide.rst), be commented (inside the PHP source files),

be documented (in the [user guide](https://codeigniter4.github.io/userguide/)), and unit tested (in the [test folder](https://github.com/codeigniter4/CodeIgniter4/tree/develop/tests)).

There is a [Contributing to CodeIgniter](./contributing/README.rst) section in the repository which describes the contribution process; this page is an overview.

Note, we expect all code changes or bug-fixes to be accompanied by one or more tests added to our test suite to prove the code works. If pull requests are not accompanied by relevant tests, they will likely be closed. Since we are a team of volunteers, we don't have any more time to work on the framework than you do. Please make it as painless for your contributions to be included as possible. If you need help with getting tests running on your local machines, ask for help on the forums. We would be happy to help out.

The [Open Source Guide](https://opensource.guide/) is a good first read for those new to contributing to open source!

## Issues

Issues are a quick way to point out a bug. If you find a bug or documentation error in CodeIgniter then please make sure that:

1. There is not already an open [Issue](https://github.com/codeigniter4/CodeIgniter4/issues)

2. The Issue has not already been fixed (check the develop branch or look for [closed Issues](https://github.com/codeigniter4/CodeIgniter4/issues?q=is%3Aissue+is%3Aclosed))

3. It's not something really obvious that you can fix yourself

Reporting Issues is helpful, but an even [better approach](./contributing/workflow.rst) is to send a [Pull Request](https://help.github.com/en/articles/creating-a-pull-request), which is done by [Forking](https://help.github.com/en/articles/fork-a-repo) the main repository and making a [Commit](https://help.github.com/en/desktop/contributing-to-projects/committing-and-reviewing-changes-to-your-project) to your own copy of the project. This will require you to use the version control system called [Git](https://git-scm.com/).

## Guidelines

Before we look into how to contribute to CodeIgniter4, here are some guidelines. If your Pull Requests fail

to pass these guidelines, they will be declined, and you will need to re-submit

when you’ve made the changes. This might sound a bit tough, but it is required

for us to maintain the quality of the codebase.

### PHP Style

All code must meet the [Style Guide](./contributing/styleguide.rst).

This makes certain that all submitted code is of the same format as the existing code and ensures that the codebase will be as readable as possible.

### Documentation

If you change anything that requires a change to documentation, then you will need to add to the documentation. New classes, methods, parameters, changing default values, etc. are all changes that require a change to documentation. Also, the [changelog](https://codeigniter4.github.io/CodeIgniter4/changelogs/index.html) must be updated for every change, and [PHPDoc](https://github.com/codeigniter4/CodeIgniter4/blob/develop/phpdoc.dist.xml) blocks must be maintained.

### Compatibility

CodeIgniter4 requires [PHP 7.2](https://php.net/releases/7\_2\_0.php).

### Branching

CodeIgniter4 uses the [Git-Flow](http://nvie.com/posts/a-successful-git-branching-model/) branching model which requires all

Pull Requests to be sent to the "develop" branch; this is where the next planned version will be developed.

The "master" branch will always contain the latest stable version and is kept clean so a "hotfix" (e.g. an

emergency security patch) can be applied to the "master" branch to create a new version, without worrying

about other features holding it up. For this reason, all commits need to be made to the "develop" branch,

and any sent to the "master" branch will be closed automatically. If you have multiple changes to submit,

please place all changes into their own branch on your fork.

\*\*One thing at a time:\*\* A pull request should only contain one change. That does not mean only one commit,

but one change - however many commits it took. The reason for this is that if you change X and Y,

but send a pull request for both at the same time, we might really want X but disagree with Y,

meaning we cannot merge the request. Using the Git-Flow branching model you can create new

branches for both of these features and send two requests.

A reminder: \*\*please use separate branches for each of your PRs\*\* - it will make it easier for you to keep changes separate from

each other and from whatever else you are doing with your repository!

### Signing

You must [GPG-sign](./contributing/signing.rst) your work, certifying that you either wrote the work or otherwise have the right to pass it on to an open-source project. This is \*not\* just a "signed-off-by" commit, but instead, a digitally signed one.

## How-to Guide

The best way to contribute is to fork the CodeIgniter4 repository, and "clone" that to your development area. That sounds like some jargon, but "forking" on GitHub means "making a copy of that repo to your account" and "cloning" means "copying that code to your environment so you can work on it".

1. Set up Git ([Windows](https://git-scm.com/download/win), [Mac](https://git-scm.com/download/mac), & [Linux](https://git-scm.com/download/linux)).

2. Go to the [CodeIgniter4 repository](https://github.com/codeigniter4/CodeIgniter4).

3. [Fork](https://help.github.com/en/articles/fork-a-repo) it (to your Github account).

4. [Clone](https://help.github.com/en/articles/cloning-a-repository) your CodeIgniter repository: `git@github.com:\<your-name>/CodeIgniter4.git`

5. Create a new [branch](https://help.github.com/en/articles/about-branches) in your project for each set of changes you want to make.

6. Fix existing bugs on the [Issue tracker](https://github.com/codeigniter4/CodeIgniter4/issues) after confirming that no one else is working on them.

7. [Commit](https://help.github.com/en/desktop/contributing-to-projects/committing-and-reviewing-changes-to-your-project) the changed files in your contribution branch.

8. [Push](https://docs.github.com/en/github/using-git/pushing-commits-to-a-remote-repository) your contribution branch to your fork.

9. Send a [pull request](https://docs.github.com/en/github/collaborating-with-issues-and-pull-requests/creating-a-pull-request-from-a-fork).

The codebase maintainers will now be alerted to the submission and someone from the team will respond. If your change fails to meet the guidelines, it will be rejected or feedback will be provided to help you improve it.

Once the maintainer handling your pull request is satisfied with it they will approve the pull request and merge it into the "develop" branch; your patch will now be part of the next release!

### Keeping your fork up-to-date

Unlike systems like Subversion, Git can have multiple remotes. A remote is the name for the URL of a Git repository. By default, your fork will have a remote named "origin", which points to your fork, but you can add another remote named "codeigniter", which points to `git://github.com/codeigniter4/CodeIgniter4.git`. This is a read-only remote, but you can pull from this develop branch to update your own.

If you are using the command-line, you can do the following to update your fork to the latest changes:

1. `git remote add codeigniter git://github.com/codeigniter4/CodeIgniter4.git`

2. `git pull codeigniter develop`

3. `git push origin develop`

Your fork is now up to date. This should be done regularly and, at the least, before you submit a pull request.

## Translations Installation

If you wish to contribute to the system message translations,

then fork and clone the [translations repository](https://github.com/codeigniter4/translations)

separately from the codebase.

These are two independent repositories!