# How to contribute

CakePHP loves to welcome your contributions. There are several ways to help out:

\* Create an [issue](https://github.com/cakephp/cakephp/issues) on GitHub, if you have found a bug

\* Write test cases for open bug issues

\* Write patches for open bug/feature issues, preferably with test cases included

\* Contribute to the [documentation](https://github.com/cakephp/docs)

There are a few guidelines that we need contributors to follow so that we have a

chance of keeping on top of things.

## Code of Conduct

Help us keep CakePHP open and inclusive. Please read and follow our [Code of Conduct](https://github.com/cakephp/code-of-conduct/blob/master/CODE\_OF\_CONDUCT.md).

## Getting Started

\* Make sure you have a [GitHub account](https://github.com/signup/free).

\* Submit an [issue](https://github.com/cakephp/cakephp/issues), assuming one does not already exist.

\* Clearly describe the issue including steps to reproduce when it is a bug.

\* Make sure you fill in the earliest version that you know has the issue.

\* Fork the repository on GitHub.

## Making Changes

\* Create a topic branch from where you want to base your work.

\* This is usually the master branch.

\* Only target release branches if you are certain your fix must be on that

branch.

\* To quickly create a topic branch based on master; `git branch

master/my\_contribution master` then checkout the new branch with `git

checkout master/my\_contribution`. Better avoid working directly on the

`master` branch, to avoid conflicts if you pull in updates from origin.

\* Make commits of logical units.

\* Check for unnecessary whitespace with `git diff --check` before committing.

\* Use descriptive commit messages and reference the #issue number.

\* Core test cases should continue to pass. You can run tests locally or enable

[travis-ci](https://travis-ci.org/) for your fork, so all tests and codesniffs

will be executed.

\* Your work should apply the [CakePHP coding standards](https://book.cakephp.org/4/en/contributing/cakephp-coding-conventions.html).

## Which branch to base the work

\* Bugfix branches will be based on master.

\* New features that are backwards compatible will be based on the appropriate 'next' branch. For example if you want to contribute to the next 3.x branch, you should base your changes on `3.next`.

\* New features or other non backwards compatible changes will go in the next major release branch. Development on 4.0 has not started yet, so breaking changes are unlikely to be merged in.

## Submitting Changes

\* Push your changes to a topic branch in your fork of the repository.

\* Submit a pull request to the repository in the CakePHP organization, with the

correct target branch.

## Test cases and codesniffer

CakePHP tests requires [PHPUnit](https://phpunit.de/manual/current/en/installation.html).

To install PHPUnit use composer:

php composer.phar require "phpunit/phpunit:\*"

To run the test cases locally use the following command:

vendor/bin/phpunit

You can copy file `phpunit.xml.dist` to `phpunit.xml` and modify the database

driver settings as required to run tests for particular database.

You can also register on [Travis CI](https://travis-ci.org/) and from your

[profile](https://travis-ci.org/profile) page enable the service hook for your

CakePHP fork on GitHub for automated test builds.

To run the sniffs for CakePHP coding standards:

vendor/bin/phpcs -p --extensions=php --standard=vendor/cakephp/cakephp-codesniffer/CakePHP ./src

Check the [cakephp-codesniffer](https://github.com/cakephp/cakephp-codesniffer)

repository to setup the CakePHP standard. The [README](https://github.com/cakephp/cakephp-codesniffer/blob/master/README.md) contains installation info

for the sniff and phpcs.

## Reporting a Security Issue

If you've found a security related issue in CakePHP, please don't open an issue in github. Instead contact us at security@cakephp.org. For more information on how we handle security issues, [see the CakePHP Security Issue Process](https://book.cakephp.org/4/en/contributing/tickets.html#reporting-security-issues).

# Additional Resources

\* [CakePHP coding standards](https://book.cakephp.org/4/en/contributing/cakephp-coding-conventions.html)

\* [Existing issues](https://github.com/cakephp/cakephp/issues)

\* [Development Roadmaps](https://github.com/cakephp/cakephp/wiki#roadmaps)

\* [General GitHub documentation](https://help.github.com/)

\* [GitHub pull request documentation](https://help.github.com/articles/creating-a-pull-request/)

\* `#cakephp` IRC channel on freenode.org