# Contributing

The easiest way to contribute is to work on a checkout of your own fork.

When working on the `core` repository, it makes sense to rename your fork to `php-telegram-bot`.

Before you contribute code, please make sure it conforms to the PSR-12 coding standard and that the unit tests still pass.

You can run the following commands to check if everything is ready to submit:

```bash

cd php-telegram-bot

composer install

composer check-code

```

Which should give you no output, indicating that there are no coding standard errors.

And then (remember to set up your test database!):

```bash

composer test

```

Which should give you no failures or errors. You can ignore any skipped tests as these are for external tools.

## Pushing

Development is based on the git flow branching model (see http://nvie.com/posts/a-successful-git-branching-model/)

If you fix a bug please push in hotfix branch.

If you develop a new feature please create a new branch.

## Version

Version number: 0.#version.#hotfix

## Further code convention adopted

- Each method and class is documented with a docblock

Example for a function or method:

```php

/\*\*

\* Get formatted date

\*

\* @param string $location

\*

\* @return string

\*/

```

- Each file is provided with the following header:

```php

/\*\*

\* This file is part of the TelegramBot package.

\*

\* (c) Avtandil Kikabidze aka LONGMAN <akalongman@gmail.com>

\*

\* For the full copyright and license information, please view the LICENSE

\* file that was distributed with this source code.

\*/

```