# Contributing

## First

\* Check if the issue you're going to submit isn't already submitted in

the [Issues](https://github.com/tmuxinator/tmuxinator/issues) page.

## Issues

\* Submit a ticket for your issue, assuming one does not already exist.

\* The issue must:

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

\* Also include all the information you can to make it easier for us to reproduce it,

like OS version, gem versions, rbenv or rvm versions etc...

\* Even better, provide a failing test case for it.

## Pull Requests

If you've gone the extra mile and have a patch that fixes the issue, you

should submit a Pull Request!

\* Please follow the [GitHub Styleguide](https://github.com/styleguide/ruby) for

Ruby in both implementation and tests!

\* Fork the repo on Github.

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

\* Add a test for your change. Only refactoring and documentation changes

require no new tests. If you are adding functionality or fixing a bug,

we need a test!

\* Run \_all\_ the tests to ensure nothing else was broken. We only take pull requests with passing tests. You can run the tests with `rake test`.

\* Make a note in the `CHANGELOG.md` file with a brief summary of your change under the heading "Unreleased" at the top of the file. If that heading does not exist, you should add it.

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

\* Structure your commit messages like this:

```

Summarize clearly in one line what the commit is about

Describe the problem the commit solves or the use

case for a new feature. Justify why you chose

the particular solution.

```

\* Use "fix", "add", "change" instead of "fixed", "added", "changed" in your commit messages.

\* Push to your fork and submit a pull request.