# Contributing

Thanks so much for considering a contribution to bootstrap\_form. We love pull requests!

We want everyone to feel welcome to contribute. We encourage respectful exchanges of ideas. We govern ourselves with the Contributor Covenant [Code of Conduct](/CODE\_OF\_CONDUCT.md).

There are a number of ways you can contribute to `bootstrap\_form`:

- Fix a bug or add a new feature

- Add to the documentation

- Review pull requests

## Code Contributions

Here's a quick guide for code contributions:

### 1. Check if issue or feature is available to work on

Make sure no one else is working on the same issue or feature. Search the issues

and pull requests for anything that looks like the issue or feature you want to

address. If no one else is working on your issue or feature, carry on with the

following steps.

### 2. (Optional) Create an issue, and wait a few days for someone to respond

If you create an issue for your feature request or bug, it gives the maintainers a chance to comment on your ideas before you invest a lot of work on a contribution. It may save you some re-work compared to simply submitting a pull request. It's up to you whether you submit an issue.

### 3. Fork the repo

Fork the project. Optionally, create a branch you want to work on.

### 4. Get it running locally

- Install the required dependencies with `bundle install`

- Run tests via: `bundle exec rake`

### 5. Hack away

- Try to keep your changes small. Consider making several smaller pull requests if your changes are extensive.

- Don't forget to add necessary tests.

- Update the README if necessary.

- Add a line to the CHANGELOG for your bug fix or feature.

- Read the [Coding Guidelines](#coding-guidelines) section and make sure that `rake lint` doesn't find any offences.

You may find the demo application useful for development and debugging.

- `cd demo`

- `rake db:schema:load`

- `rails s`

- Navigate to http://localhost:3000

### 6. Make a pull request

- If you've never made a pull request (PR) before, read this: https://help.github.com/articles/about-pull-requests/.

- If your PR fixes an issues, be sure to put "Fixes #nnn" in the description of the PR (where `nnn` is the issue number). Github will automatically close the issue when the PR is merged.

- When the PR is submitted, check if Travis CI ran all the tests successfully, and didn't raise any issues.

### 7. Done!

Somebody will shortly review your pull request and if everything is good, it will be

merged into the master branch. Eventually the gem will be published with your changes.

### Coding guidelines

This project uses [RuboCop](https://github.com/bbatsov/rubocop) to enforce standard Ruby coding

guidelines.

\* Test that your contribution passes with `rake rubocop`.

\* RuboCop is also run as part of the full test suite with `bundle exec rake`.

\* Note the Travis build will fail and your PR cannot be merged if RuboCop finds offences.

Note that most editors have plugins to run RuboCop as you type, or when you save a file. You may find it well worth your time to install and configure the RuboCop plugin for your editor. Read the [RuboCop documentation](https://rubocop.readthedocs.io/en/latest/integration\_with\_other\_tools/).

### Supported Versions of Ruby and Rails

The goal of `bootstrap\_form` is to support all versions of Rails currently supported for bug fixes and security issues. We do not test against versions supported for severe security issues. We test against the minimum [version of Ruby required](https://guides.rubyonrails.org/upgrading\_ruby\_on\_rails.html#ruby-versions) for those versions of Rails.

The Ruby on Rails support policy is [here](https://guides.rubyonrails.org/maintenance\_policy.html).

## Documentation Contributions

Contributions to documentation are always welcome. Even fixing one typo improves the quality of `bootstrap\_form`. To make a documentation contribution, follow steps 1-3 of Code Contributions, then make the documentation changes, then make the pull request (step 6 of Code Contributions).

If you put `[ci skip]` in the commit message of the most recent commit of the PR, you'll be a good citizen by not causing Travis CI to run all the tests when it's not necessary.

## Reviewing Pull Requests

We are an entirely volunteer project. Sometimes it's hard for people to find the time to review pull requests. You can help! If you see a pull request that's waiting to be merged, it could be because no one has reviewed it yet. Your review could help move the pull request forward to be merged.

---

Thanks to all the great contributors over the years: https://github.com/bootstrap-ruby/bootstrap\_form/graphs/contributors

## Troubleshooting

### Models and Database Tables

`bootstrap\_form` needs a few models and tables to support testing. It appears that the necessary tables were created via the `demo/db/schema.rb` file. To support `rich\_text\_area`, Rails 6 creates some migrations. These migrations had to be run in the existing database (not an empty one) to create a new `schema.rb` that creates the `bootstrap\_form` test tables, and the tables needed by Rails 6. The `schema.rb` file was checked in to GitHub, but the migrations were not.

In the future, any new Rails functionality that creates tables would likely have to be prepared the same way:

```

cd demo

rails db:setup # create the databases from `schema.rb`

rails db:migrate # add the new tables and create a new `schema.rb`

```