# Contributing to Verdaccio

We are happy that you wish to contribute to this project. For that reason, we

present you with this guide.

Additional information is available on the

[wiki](https://github.com/verdaccio/verdaccio/wiki).

## Contents

- [Contents](#contents)

- [How Do I Contribute?](#how-do-i-contribute)

- [Development](#development)

- [Reporting Bugs](#reporting-bugs)

- [Issue Search](#issue-search)

- [Check Website For Solution](#check-website-for-solution)

- [Chat](#chat)

- [Check If It's Been Fixed](#check-if-its-been-fixed)

- [Request Features](#request-features)

- [Submitting a Pull Request](#submitting-a-pull-request)

- [Make Changes and Commit](#make-changes-and-commit)

- [Update Tests](#update-tests)

- [Develop Plugins](#develop-plugins)

## How Do I Contribute?

There are different ways to contribute, each with a different level

of involvement and technical knowledge required, such as:

\* [Reporting Bugs](#reporting-bugs)

\* [Request Features](#request-features)

\* [Develop Plugins](#develop-plugins)

\* [Improve Documentation](http://www.verdaccio.org/docs/en/installation.html)

\*\*Please read this document carefully. It will help maintainers and readers

in solving your issue(s), evaluating your feature request, etc.\*\*

## Development

Development guides can be found on the [wiki](https://github.com/verdaccio/verdaccio/wiki):

\* [Building the project](https://github.com/verdaccio/verdaccio/wiki/Build-Source-Code)

\* [Running, debugging, and testing](https://github.com/verdaccio/verdaccio/wiki/Running-and-Debugging-tests)

## Reporting Bugs

We welcome clear, detailed bug reports.

\*\*Bugs are considered features that are not working as described in

documentation.\*\*

If you've found a bug in Verdaccio \*\*that isn't a security risk\*\*, please file

a report in our [issue tracker](https://github.com/verdaccio/verdaccio/issues).

\*\*NOTE: Verdaccio still does not support all npm commands. Some were not

considered important and others have not been requested yet.\*\*

### Issue Search

Search to see if it has already been reported via

the issue search.

Additionally, we have labelled questions for easy follow-up as [questions](https://github.com/verdaccio/verdaccio/labels/question).

If so, up-vote it (using GitHub reactions) or add additional helpful details to

the existing issue to show that it's affecting multiple people.

### Check Website For Solution

Some of the most popular topics can be found in our website(http://www.verdaccio.org/docs/en/installation.html)

### Chat

Questions can be asked via [Discord](http://chat.verdaccio.org/)

\*\*Please use the `#questions#` and `#development` channels.\*\*

### Check If It's Been Fixed

Check if the issue has been fixed — try to reproduce it using the latest

`master` or development branch in the repository.

## Request Features

New feature requests are welcome. Analyse whether the idea fits within scope of

the project. Then, detail your request, ensuring context and use case is provided.

\*\*Please provide:\*\*

\* A detailed description the advantages of your request

\* Whether or not it's compatible with `npm` and `yarn`

\* A potential implementation or design

\* Whatever else you have in your mind ?

### Submitting a Pull Request

The following are the steps you should follow when creating a pull request.

Subsequent pull requests only need to follow step 3 and beyond.

1. Fork the repository on GitHub

2. Clone the forked repository to your machine

3. Make your changes and commit them to your local repository

4. Rebase and push your commits to your GitHub remote fork/repository

5. Issue a Pull Request to the official repository

6. Your Pull Request is reviewed by a committer and merged into the repository

\*\*NOTE\*\*: While there are other ways to accomplish the steps using other tools,

the examples here will assume most actions will be performed via `git` on

command line.

For more information on maintaining a fork, please see the GitHub Help article

titled [Fork a Repo](https://help.github.com/articles/fork-a-repo/), and

information on [rebasing](https://git-scm.com/book/en/v2/Git-Branching-Rebasing).

### Make Changes and Commit

#### Before Commit

Before committing, \*\*you must ensure there are no linting errors and

all tests pass.\*\*

To do this, run all tests (including e2e):

```bash

yarn test:all

```

Then, and only then, you can create your pull request.

#### Commit Guidelines

We follow the [conventional commit messages](https://conventionalcommits.org/)

convention in order to automate CHANGELOG generation and to automate

semantic versioning.

For example:

\* `feat: A new feature`

\* `fix: A bug fix`

A commit of the type feat introduces a new feature to the codebase

(this correlates with MINOR in semantic versioning).

e.g.:

```

feat: xxxxxxxxxx

```

A commit of the type fix patches a bug in your codebase (this correlates with PATCH in semantic versioning).

e.g.:

```

fix: xxxxxxxxxx

```

Commits types such as as `docs:`,`style:`,`refactor:`,`perf:`,`test:`

and `chore:` are valid but have no effect on versioning. \*\*It would be great if you use them.\*\*

All commits message are going to be validated when they are created using husky hooks.

\*\*PRs that do not follow the commit message guidelines will not be merged.\*\*

## Update Tests

\*\*Any change in source code must include test updates\*\*.

If you need help with how testing works, please [refer to the following guide](https://github.com/verdaccio/verdaccio/wiki/Running-and-Debugging-tests).

\*\*If you are introducing new features, you MUST include new tests. PRs for

features without tests will not be merged.\*\*

Things excluded from tests:

\* Documentation

\* Website

\* Build

\* Deployment

\* Assets

\* Flow types

## Develop Plugins

Plugins are add-ons that extend the functionality of the application.

If you want to develop your own plugin:

1. Check whether there is a legacy Sinopia plugin for the feature that you need

via [npmjs](https://www.npmjs.com/search?q=sinopia)

2. Keep in mind the [life-cycle to load a plugin](https://verdaccio.org/docs/en/dev-plugins)

3. You are free to host your plugin in your repository or ours (just ask)

4. Provide a detailed description of your plugin to help users understand it