# Contribute to Strapi

Strapi is an open-source project administered by [the Strapi team](https://strapi.io/company). We appreciate your interest and efforts to contribute to Strapi.

All efforts to contribute are highly appreciated, we recommend you talk to a maintainer prior to spending a lot of time making a pull request that may not align with the project roadmap.

## Open Development & Community Driven

Strapi is an open-source project. See the [LICENSE](https://github.com/strapi/strapi/blob/master/LICENSE) file for licensing information. All the work done is available on GitHub.

The core team and the contributors send pull requests which go through the same validation process.

## Feature Requests

Feature Requests by the community are highly encouraged. Please feel free to submit a [feature request](https://portal.productboard.com/strapi) or to upvote ? [an existing feature request](https://portal.productboard.com/strapi) in the ProductBoard.

## RFCs

Some important changes in Strapi require some thoughts to be put into the design phase before starting working on a PR.

The RFC (Request For Comments) process will help us create consensus among the core team and include as much feedback as possible from the community, for these upcoming changes.

Before contributing, you will probably have to create a RFC on this [strapi/rfcs](https://github.com/strapi/rfcs) repo.

## Code of Conduct

This project and everyone participating in it are governed by the [Strapi Code of Conduct](CODE\_OF\_CONDUCT.md). By participating, you are expected to uphold this code. Please read the [full text](CODE\_OF\_CONDUCT.md) so that you can read which actions may or may not be tolerated.

## Documentation

Pull requests relating to fixing documentation for the latest release should be directed towards the [documentation branch](https://github.com/strapi/strapi/tree/documentation) \*\*not\*\* towards the master branch. Any PRs made towards the master branch will not be released until the next Strapi version release.

## Bugs

We are using [GitHub Issues](https://github.com/strapi/strapi/issues) to manage our public bugs. We keep a close eye on this so before filing a new issue, try to make sure the problem does not already exist.

---

## Before Submitting a Pull Request

The core team will review your pull request and will either merge it, request changes to it, or close it.

\*\*Before submitting your pull request\*\* make sure the following requirements are fulfilled:

- Fork the repository and create your branch from `master`.

- Run `yarn setup` in the repository root.

- If you’ve fixed a bug or added code that should be tested, add the tests and then link the corresponding issue in either your commit or your PR!

- Ensure the test suites are passing:

- `yarn test:unit`

- `yarn test:front`

- Make sure your code lints (`yarn lint`).

## Contribution Prerequisites

- You have [Node](https://nodejs.org/en/) at v10.10.0+ only and [Yarn](https://yarnpkg.com/en/) at v1.2.0+.

- You are familiar with Git.

## Development Workflow

To facilitate the contribution, we have drastically reduced the amount of commands necessary to install the entire development environment.

First of all, you need to check if you're using the [required versions of Node.js and npm](https://strapi.io/documentation/v3.x/installation/cli.html#step-1-make-sure-requirements-are-met)

Then, please follow the instructions below:

#### 1. Fork the [repository](https://github.com/strapi/strapi)

[Go to the repository](https://github.com/strapi/strapi) and fork it to your own GitHub account.

#### 2. Clone from your repository

```bash

git clone git@github.com:YOUR\_USERNAME/strapi.git

```

#### 3. Install the dependencies

Go to the root of the repository.

```bash

cd strapi && yarn setup

```

#### 4. Start the example application

Read the `getstarted` application README [here](./examples/getstarted/README.md).

#### 5. Running the administration panel in development mode

\*\*Start the administration panel server for development\*\*

```bash

cd strapi/packages/strapi-admin

yarn develop

```

The administration panel will be available at http://localhost:4000/admin

\*\*Awesome! You are now able to contribute to Strapi.\*\*

#### 5. Available commands

- `yarn watch` starts yarn watch in all packages.

- `yarn build` builds the `strapi-helper-plugin` (use this command when you develop in the administration panel).

- `yarn setup` installs the dependencies.

- `yarn lint` lints the codebase.

- `yarn test:clean` removes the coverage.

- `yarn test:front` runs the front-end related tests.

- `yarn test:front:watch` runs an interactive test watcher for the front-end.

- `yarn test:snyk` checks the dependencies vulnerabilities.

- `yarn test:unit` runs the back-end unit tests.

- `yarn test:e2e` runs an end-to-end test suite.

- `yarn test:generate-app` generates a test application.

- `yarn test:start-app` starts the test application.

---

## Running the tests

\*\*Changing the database:\*\*

You can run the test suites using different databases:

```bash

$ node test/e2e.js --db=sqlite

$ node test/e2e.js --db=mongo

$ node test/e2e.js --db=postgres

$ node test/e2e.js --db=mysql

```

---

## Miscellaneous

### Repository Organization

We chose to to use a monorepo design that exploits [Yarn Workspaces](https://yarnpkg.com/en/docs/workspaces) in the way [React](https://github.com/facebook/react/tree/master/packages) or [Babel](https://github.com/babel/babel/tree/master/packages) does. This allows the community to easily maintain the whole ecosystem, keep it up-to-date and consistent.

We do our best to keep the master branch as clean as possible, with tests passing at all times. However, it may happen that the master branch moves faster than the release cycle. Therefore check the [releases on npm](https://www.npmjs.com/package/strapi) so that you're always up-to-date with the latest stable version.

### Reporting an issue

Before submitting an issue you need to make sure:

- You are experiencing a concrete technical issue with Strapi.

- You have already searched for related [issues](https://github.com/strapi/strapi/issues), and found none open (if you found a related \_closed\_ issue, please link to it from your post).

- You are not asking a question about how to use Strapi or about whether or not Strapi has a certain feature. For general help using Strapi, you may:

- Refer to [the official Strapi documentation](http://strapi.io).

- Ask a member of the community in the [Strapi Slack Community](https://slack.strapi.io/).

- Ask a question on [Github Discussions](https://github.com/strapi/strapi/discussions).

- Your issue title is concise, on-topic and polite.

- You can and do provide steps to reproduce your issue.

- You have tried all the following (if relevant) and your issue remains:

- Make sure you have the right application started.

- Make sure the [issue template](.github/ISSUE\_TEMPLATE) is respected.

- Make sure your issue body is readable and [well formatted](https://guides.github.com/features/mastering-markdown).

- Make sure you've killed the Strapi server with CTRL+C and started it again.

- Make sure the application you are using to reproduce the issue has a clean `node\_modules` directory, meaning:

- no dependencies are linked (e.g. you haven't run `npm link`)

- that you haven't made any inline changes to files in the `node\_modules` folder

- that you don't have any weird global dependency loops. The easiest way to double-check any of the above, if you aren't sure, is to run: `$ rm -rf node\_modules && npm cache clear && npm install`.