# How to Contribute

Jest is one of Facebook's open source projects that is both under very active development and is also being used to ship code to everybody on [facebook.com](https://www.facebook.com). We're still working out the kinks to make contributing to this project as easy and transparent as possible, but we're not quite there yet. Hopefully this document makes the process for contributing clear and answers some questions that you may have.

## [Code of Conduct](https://code.facebook.com/codeofconduct)

Facebook has adopted a Code of Conduct that we expect project participants to adhere to. Please read [the full text](https://code.facebook.com/codeofconduct) so that you can understand what actions will and will not be tolerated.

## Open Development

All work on Jest happens directly on [GitHub](/). Both core team members and external contributors send pull requests which go through the same review process.

### `master` is unsafe

We will do our best to keep `master` in good shape, with tests passing at all times. But in order to move fast, we will make API changes that your application might not be compatible with. We will do our best to communicate these changes and always version appropriately so you can lock into a specific version if need be.

### Workflow and Pull Requests

The core team will be monitoring for pull requests. When we get one, we'll run some Facebook-specific integration tests on it first. From here, we'll need to get another person to sign off on the changes and then merge the pull request. For API changes we may need to fix internal uses, which could cause some delay. We'll do our best to provide updates and feedback throughout the process.

\_Before\_ submitting a pull request, please make sure the following is done…

1. Fork the repo and create your branch from `master`. A guide on how to fork a repository: https://help.github.com/articles/fork-a-repo/

Open terminal (e.g. Terminal, iTerm, Git Bash or Git Shell) and type:

```sh-session

$ git clone https://github.com/<your\_username>/jest

$ cd jest

$ git checkout -b my\_branch

```

Note: Replace `<your\_username>` with your GitHub username

1. Jest uses [Yarn](https://code.facebook.com/posts/1840075619545360) for running development scripts. If you haven't already done so, please [install yarn](https://yarnpkg.com/en/docs/install).

1. Make sure you have `python` installed. Python is required by [node-gyp](https://github.com/nodejs/node-gyp) that is used when running `yarn install`.

To check your version of Python and ensure it's installed you can type:

```sh

python --version

```

1. Make sure you have a compatible version of `node` installed (As of October 25th 2019, `v12.x` is recommended).

```sh

node -v

```

1. Run `yarn install`. On Windows: To install [Yarn](https://yarnpkg.com/en/docs/install#windows-tab) on Windows you may need to download either node.js or Chocolatey<br />

```sh

yarn install

```

To check your version of Yarn and ensure it's installed you can type:

```sh

yarn --version

```

On Windows `yarn install` may fail with `gyp ERR! build error`. One of possible solutions:

```sh

yarn global add windows-build-tools

```

1. Run `yarn build` to transpile TypeScript to JavaScript and type check the code

```sh

yarn build

```

1. If you've added code that should be tested, add tests. You can use watch mode that continuously transforms changed files to make your life easier.

```sh

# in the background

yarn watch

```

1. If you've changed APIs, update the documentation.

1. Ensure the test suite passes via `yarn jest`. To run the test suite you may need to install [Mercurial](https://www.mercurial-scm.org/) (`hg`). On macOS, this can be done using [homebrew](http://brew.sh/): `brew install hg`.

```sh-session

$ brew install hg # maybe

$ yarn test

```

1. If you haven't already, complete the [CLA](https://code.facebook.com/cla/).

#### Changelog entries

All changes that add a feature to or fix a bug in any of Jest's packages require a changelog entry containing the names of the packages affected, a description of the change, and the number of and link to the pull request. Try to match the structure of the existing entries.

For significant changes to the documentation or website and things like cleanup, refactoring, and dependency updates, the "Chore & Maintenance" section of the changelog can be used.

You can add or edit the changelog entry in the GitHub web interface once you have opened the pull request and know the number and link to it.

Make sure to alphabetically order your entry based on package name. If you have changed multiple packages, separate them with a comma.

#### Testing

Code that is written needs to be tested to ensure that it achieves the desired behaviour. Tests either fall into a unit test or an integration test.

##### Unit tests

Some of the packages within jest have a `\_\_tests\_\_` directory. This is where unit tests reside in. If the scope of your work only requires a unit test, this is where you will write it in. Tests here usually don't require much if any setup.

##### Integration tests

There will be situations however where the work you have done cannot be tested alone using unit tests. In situations like this, you should write an integration test for your code. The integration tests reside within the `e2e` directory. Within this directory, there is a `\_\_tests\_\_` directory. This is where you will write the integration test itself. The tests within this directory execute jest itself using `runJest.js` and assertions are usually made on one if not all the output of the following `status`, `stdout` and `stderr`. The other sub directories within the `e2e` directory are where you will write the files that jest will run for your integration tests. Feel free to take a look at any of the tests in the `\_\_tests\_\_` directory within `e2e` to have a better sense of how it is currently being done.

It is possible to run the integration test itself manually to inspect that the new behaviour is indeed correct. Here is a small code snippet of how to do just that. This is useful when debugging a failing test.

```bash

$ cd e2e/clear-cache

$ node ../../packages/jest-cli/bin/jest.js # It is possible to use node --inspect or ndb

PASS \_\_tests\_\_/clear\_cache.test.js

? stub (3ms)

Test Suites: 1 passed, 1 total

Tests: 1 passed, 1 total

Snapshots: 0 total

Time: 0.232 s, estimated 1 s

Ran all test suites.

```

##### Using jest-circus

There may be cases where you want to run jest using `jest-circus` instead of `jest-jasmine2` (which is the default runner) for integration testing. In situations like this, set the environment variable `JEST\_CIRCUS` to 1. That will configure jest to use `jest-circus`. So something like this.

```bash

JEST\_CIRCUS=1 yarn jest

```

#### Additional Workflow for any changes made to website or docs

If you are making changes to the website or documentation, test the website folder and run the server to check if your changes are being displayed accurately.

1. Locate to the website directory and install any website specific dependencies by typing in `yarn`. Following steps are to be followed for this purpose from the root directory.

```sh-session

$ cd website # Only needed if you are not already in the website directory

$ yarn

$ yarn start

```

1. You can run a development server to check if the changes you made are being displayed accurately by running `yarn start` in the website directory.

The Jest website also offers documentation for older versions of Jest, which you can edit in `website/versioned\_docs`. After making changes to the current documentation in `docs`, please check if any older versions of the documentation have a copy of the file where the change is also relevant and apply the changes to the `versioned\_docs` as well.

### Contributor License Agreement (CLA)

In order to accept your pull request, we need you to submit a CLA. You only need to do this once, so if you've done this for another Facebook open source project, you're good to go. If you are submitting a pull request for the first time, just let us know that you have completed the CLA and we can cross-check with your GitHub username.

[Complete your CLA here.](https://code.facebook.com/cla)

## How to try a development build of Jest in another project

To link `jest` on the command line to `jest-cli/bin/jest.js` in a development build:

```sh-session

$ cd /path/to/your/Jest\_clone/packages/jest-cli

$ yarn link

```

To build Jest:

```sh-session

$ cd /path/to/your/Jest\_clone

# Do one of the following:

# Check out a commit from another contributor, and then

$ yarn run build

# Or, save your changes to Jest, and then

$ yarn test # which also builds Jest

```

To run tests in another project with the development build of Jest:

```sh-session

$ cd /path/to/another/project

# link development build to the other project

$ yarn link jest-cli

$ jest [options] # run jest-cli/bin/jest.js in the development build

```

- To decide whether to specify any options, see `test` under `scripts` in the `package.json` file of the other project.

To unlink `jest` on the command line from `jest-cli/bin/jest.js` in a development build:

```sh

yarn unlink jest-cli

```

## Bugs

### Where to Find Known Issues

We will be using GitHub Issues for our public bugs. We will keep a close eye on this and try to make it clear when we have an internal fix in progress. Before filing a new issue, try to make sure your problem doesn't already exist.

### Reporting New Issues

The best way to get your bug fixed is to provide a reduced test case. Please provide a public repository with a runnable example.

### Docs translation

We get translations from crowdin, see https://crowdin.com/project/jest. Any and all help is very much appreciated!

### Security Bugs

Facebook has a [bounty program](https://www.facebook.com/whitehat/) for the safe disclosure of security bugs. With that in mind, please do not file public issues; go through the process outlined on that page.

## How to Get in Touch

`#testing` on [Reactiflux](https://www.reactiflux.com/)

## Code Conventions

- 2 spaces for indentation (no tabs).

- 80 character line length strongly preferred.

- Prefer `'` over `"`.

- ES6 syntax when possible.

- Use [TypeScript](https://www.typescriptlang.org/).

- Use semicolons;

- Trailing commas,

- Avd abbr wrds.

## Credits

This project exists thanks to all the people who [contribute](CONTRIBUTING.md).

<a href="graphs/contributors"><img src="https://opencollective.com/jest/contributors.svg?width=890&button=false" /></a>

### [Backers](https://opencollective.com/jest#backer)

Thank you to all our backers! ?

<a href="https://opencollective.com/jest#backers" target="\_blank"><img src="https://opencollective.com/jest/backers.svg?width=890"></a>

### [Sponsors](https://opencollective.com/jest#sponsor)

Support this project by becoming a sponsor. Your logo will show up here with a link to your website.

<a href="https://opencollective.com/jest/sponsor/0/website" target="\_blank"><img src="https://opencollective.com/jest/sponsor/0/avatar.svg"></a> <a href="https://opencollective.com/jest/sponsor/1/website" target="\_blank"><img src="https://opencollective.com/jest/sponsor/1/avatar.svg"></a> <a href="https://opencollective.com/jest/sponsor/2/website" target="\_blank"><img src="https://opencollective.com/jest/sponsor/2/avatar.svg"></a> <a href="https://opencollective.com/jest/sponsor/3/website" target="\_blank"><img src="https://opencollective.com/jest/sponsor/3/avatar.svg"></a> <a href="https://opencollective.com/jest/sponsor/4/website" target="\_blank"><img src="https://opencollective.com/jest/sponsor/4/avatar.svg"></a> <a href="https://opencollective.com/jest/sponsor/5/website" target="\_blank"><img src="https://opencollective.com/jest/sponsor/5/avatar.svg"></a> <a href="https://opencollective.com/jest/sponsor/6/website" target="\_blank"><img src="https://opencollective.com/jest/sponsor/6/avatar.svg"></a> <a href="https://opencollective.com/jest/sponsor/7/website" target="\_blank"><img src="https://opencollective.com/jest/sponsor/7/avatar.svg"></a> <a href="https://opencollective.com/jest/sponsor/8/website" target="\_blank"><img src="https://opencollective.com/jest/sponsor/8/avatar.svg"></a> <a href="https://opencollective.com/jest/sponsor/9/website" target="\_blank"><img src="https://opencollective.com/jest/sponsor/9/avatar.svg"></a>

## License

By contributing to Jest, you agree that your contributions will be licensed under its MIT license.