# Contributing to React Native

<!-- generated\_contributing\_start -->

React Native is one of Facebook's first open source projects that is both under very active development and is also being used to ship code to everybody using Facebook's mobile apps. If you're interested in contributing to React Native, hopefully this document makes the process for contributing clear.

## [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.

## Get involved

There are many ways to contribute to React Native, and many of them do not involve writing any code. Here's a few ideas to get started:

\* Simply start using React Native. Go through the [Getting Started](https://facebook.github.io/react-native/docs/getting-started.html) guide. Does everything work as expected? If not, we're always looking for improvements. Let us know by [opening an issue](https://facebook.github.io/react-native/docs/contributing.html#reporting-new-issues).

\* Look through the [open issues](https://github.com/facebook/react-native/issues). Provide workarounds, ask for clarification, or suggest labels. Help [triage issues](https://facebook.github.io/react-native/docs/contributing.html#triaging-issues-and-pull-requests).

\* If you find an issue you would like to fix, [open a pull request](https://facebook.github.io/react-native/docs/contributing.html#your-first-pull-request). Issues tagged as [\_Good first issue\_](https://github.com/facebook/react-native/labels/Good%20first%20issue) are a good place to get started.

\* Read through the [React Native docs](https://facebook.github.io/react-native/docs/getting-started.html). If you find anything that is confusing or can be improved, you can make edits by clicking "Improve this page" at the bottom of most docs.

\* Browse [Stack Overflow](https://stackoverflow.com/questions/tagged/react-native) and answer questions. This will help you get familiarized with common pitfalls or misunderstandings, which can be useful when contributing updates to the documentation.

\* Take a look at the [features requested](https://react-native.canny.io/feature-requests) by others in the community and consider opening a pull request if you see something you want to work on.

Contributions are very welcome. If you think you need help planning your contribution, please hop into [#react-native](https://discord.gg/0ZcbPKXt5bZjGY5n) and let people know you're looking for a mentor.

Core contributors to React Native meet monthly and post their meeting notes on the [React Native blog](https://facebook.github.io/react-native/blog). You can also find ad hoc discussions in the [React Native Core Contributors](https://www.facebook.com/groups/reactnativeoss/) Facebook group.

### Triaging issues and pull requests

One great way you can contribute to the project without writing any code is to help triage issues and pull requests as they come in.

\* Ask for more information if the issue does not provide all the details required by the template.

\* Suggest [labels](https://github.com/facebook/react-native/labels) that can help categorize issues.

\* Flag issues that are stale or that should be closed.

\* Ask for test plans and review code.

You can learn more about handling issues in the [maintainer's guide](docs/maintainers.html#handling-issues).

## Our development process

Some of the core team will be working directly on [GitHub](https://github.com/facebook/react-native). These changes will be public from the beginning. Other changesets will come via a bridge with Facebook's internal source control. This is a necessity as it allows engineers at Facebook outside of the core team to move fast and contribute from an environment they are comfortable in.

When a change made on GitHub is approved, it will first be imported into Facebook's internal source control. The change will eventually sync back to GitHub as a single commit once it has passed all internal tests.

### Branch organization

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](https://github.com/facebook/react-native/releases) and version appropriately so you can lock into a specific version if need be.

To see what changes are coming and provide better feedback to React Native contributors, use the [latest release candidate](https://facebook.github.io/react-native/versions.html) when possible. By the time a release candidate is released, the changes it contains will have been shipped in production Facebook apps for over two weeks.

## Bugs

We use [GitHub Issues](https://github.com/facebook/react-native/issues) for our public bugs. If you would like to report a problem, take a look around and see if someone already opened an issue about it. If you a are certain this is a new, unreported bug, you can submit a [bug report](https://facebook.github.io/react-native/docs/contributing.html#reporting-new-issues).

If you have questions about using React Native, the [Community page](https://facebook.github.io/react-native/help.html) list various resources that should help you get started.

We also have a [place where you can request features or enhancements](https://react-native.canny.io/feature-requests). If you see anything you'd like to be implemented, vote it up and explain your use case.

## Reporting new issues

When [opening a new issue](https://github.com/facebook/react-native/issues/new), always make sure to fill out the [issue template](https://raw.githubusercontent.com/facebook/react-native/master/.github/ISSUE\_TEMPLATE.md). \*\*This step is very important!\*\* Not doing so may result in your issue getting closed. Don't take this personally if this happens, and feel free to open a new issue once you've gathered all the information required by the template.

\* \*\*One issue, one bug:\*\* Please report a single bug per issue.

\* \*\*Provide a Snack:\*\* The best way to get attention on your issue is to provide a reduced test case. You can use [Snack](https://snack.expo.io/) to demonstrate the issue.

\* \*\*Provide reproduction steps:\*\* List all the steps necessary to reproduce the issue. Provide a Snack or upload a sample project to GitHub. The person reading your bug report should be able to follow these steps to reproduce your issue with minimal effort.

\* \*\*Try out the latest version:\*\* Verify that the issue can be reproduced locally by updating your project to use [React Native from `master`](https://facebook.github.io/react-native/versions.html). The bug may have already been fixed!

We're not able to provide support through GitHub Issues. If you're looking for help with your code, consider asking on [Stack Overflow](https://stackoverflow.com/questions/tagged/react-native) or reaching out to the community through [other channels](https://facebook.github.io/react-native/support.html).

### 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.

## Pull requests

### Your first pull request

So you have decided to contribute code back to upstream by opening a pull request. You've invested a good chunk of time, and we appreciate it. We will do our best to work with you and get the PR looked at.

Working on your first Pull Request? You can learn how from this free video series:

[\*\*How to Contribute to an Open Source Project on GitHub\*\*](https://egghead.io/courses/how-to-contribute-to-an-open-source-project-on-github)

We have a list of [beginner friendly issues](https://github.com/facebook/react-native/labels/Good%20First%20Task) to help you get your feet wet in the React Native codebase and familiar with our contribution process. This is a great place to get started.

### Proposing a change

If you would like to request a new feature or enhancement but are not yet thinking about opening a pull request, we have a [place to track feature requests](https://react-native.canny.io/feature-requests).

If you intend to change the public API, or make any non-trivial changes to the implementation, we recommend [filing an issue](https://github.com/facebook/react-native/issues/new?title=%5BProposal%5D) that includes `[Proposal]` in the title. This lets us reach an agreement on your proposal before you put significant effort into it. These types of issues should be rare. If you have been contributing to the project long enough, you will probably already have access to the [React Native Core Contributors](https://www.facebook.com/groups/reactnativeoss/) Facebook Group, where this sort of discussion is usually held.

If you're only fixing a bug, it's fine to submit a pull request right away but we still recommend to file an issue detailing what you're fixing. This is helpful in case we don't accept that specific fix but want to keep track of the issue.

### Sending a pull request

Small pull requests are much easier to review and more likely to get merged. Make sure the PR does only one thing, otherwise please split it.

Please make sure the following is done when submitting a pull request:

1. Fork [the repository](https://github.com/facebook/react-native) and create your branch from `master`.

2. Add the copyright notice to the top of any new files you've added.

3. Describe your [\*\*test plan\*\*](https://facebook.github.io/react-native/docs/contributing.html#test-plan) in your pull request description. Make sure to [test your changes](https://facebook.github.io/react-native/docs/testing.html)!

4. Make sure your code lints (`npm run lint`).

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

All pull requests should be opened against the `master` branch. After opening your pull request, ensure [\*\*all tests pass\*\*](https://facebook.github.io/react-native/docs/contributing.html#contrinuous-integration-tests) on Circle CI. If a test fails and you believe it is unrelated to your change, leave a comment on the pull request explaining why.

> \*\*Note:\*\* It is not necessary to keep clicking `Merge master to your branch` on the PR page. You would want to merge master if there are conflicts or tests are failing. The Facebook-GitHub-Bot ultimately squashes all commits to a single one before merging your PR.

#### Test plan

A good test plan has the exact commands you ran and their output, provides screenshots or videos if the pull request changes UI.

\* If you've added code that should be tested, add tests!

\* If you've changed APIs, update the documentation.

See [What is a Test Plan?](https://medium.com/@martinkonicek/what-is-a-test-plan-8bfc840ec171#.y9lcuqqi9) to learn more.

#### Continuous integration tests

Make sure all \*\*tests pass\*\* on [Circle CI][circle]. PRs that break tests are unlikely to be merged. Learn more about [testing your changes here](https://facebook.github.io/react-native/docs/testing.html).

[circle]: https://circleci.com/gh/facebook/react-native

#### Breaking changes

When adding a new breaking change, follow this template in your pull request:

```

### New breaking change here

\* \*\*Who does this affect\*\*:

\* \*\*How to migrate\*\*:

\* \*\*Why make this breaking change\*\*:

\* \*\*Severity (number of people affected x effort)\*\*:

```

If your pull request is merged, a core contributor will update the [list of breaking changes](https://github.com/facebook/react-native/wiki/Breaking-Changes) which is then used to populate the release notes.

#### Copyright Notice for files

Copy and paste this to the top of your new file(s):

```JS

/\*\*

\* Copyright (c) 2015-present, Facebook, Inc.

\* All rights reserved.

\*

\* This source code is licensed under the BSD-style license found in the

\* LICENSE file in the root directory of this source tree. An additional grant

\* of patent rights can be found in the PATENTS file in the same directory.

\*/

```

If you've added a new module, add a `@providesModule <moduleName>` at the end of the comment. This will allow the haste package manager to find it.

#### 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, the Facebook GitHub Bot will reply with a link to the CLA form. You may also [complete your CLA here](https://code.facebook.com/cla).

### What happens next?

The core team will be monitoring for pull requests. Read [what to expect from maintainers](https://facebook.github.io/react-native/docs/maintainers.html#handling-pull-requests) to understand what may happen after you open a pull request.

## Style Guide

Our linter will catch most styling issues that may exist in your code. You can check the status of your code styling by simply running `npm run lint`.

However, there are still some styles that the linter cannot pick up.

### Code Conventions

#### General

\* \*\*Most important: Look around.\*\* Match the style you see used in the rest of the project. This includes formatting, naming things in code, naming things in documentation.

\* Add trailing commas,

\* 2 spaces for indentation (no tabs)

\* "Attractive"

#### JavaScript

\* Use semicolons;

\* ES6 standards

\* Prefer `'` over `"`

\* Do not use the optional parameters of `setTimeout` and `setInterval`

\* 80 character line length

#### JSX

\* Prefer `"` over `'` for string literal props

\* When wrapping opening tags over multiple lines, place one prop per line

\* `{}` of props should hug their values (no spaces)

\* Place the closing `>` of opening tags on the same line as the last prop

\* Place the closing `/>` of self-closing tags on their own line and left-align them with the opening `<`

#### Objective-C

\* Space after `@property` declarations

\* Brackets on \*every\* `if`, on the \*same\* line

\* `- method`, `@interface`, and `@implementation` brackets on the following line

\* \*Try\* to keep it around 80 characters line length (sometimes it's just not possible...)

\* `\*` operator goes with the variable name (e.g. `NSObject \*variableName;`)

#### Java

\* If a method call spans multiple lines closing bracket is on the same line as the last argument.

\* If a method header doesn't fit on one line each argument goes on a separate line.

\* 100 character line length

### Documentation

\* Do not wrap lines at 80 characters - configure your editor to soft-wrap when editing documentation.

## License

By contributing to React Native, you agree that your contributions will be licensed under its BSD license.

<!-- generated\_contributing\_end -->