# Contributing to React Native Modal

## Development Process

All work on React Native Modal happens directly on GitHub. Contributors send pull requests which go through a review process.

> \*\*Working on your first pull request?\*\* You can learn how from this \_free\_ series: [How to Contribute to an Open Source Project on GitHub](https://egghead.io/series/how-to-contribute-to-an-open-source-project-on-github).

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/)).

2. Run `yarn` or `npm install` to install all required dependencies.

3. Now you are ready to make your changes!

### Development flow

- Head into the `example` folder and do the same.

- Head into `ios` and run `pod install` if you wish to compile under iOS

- Get back into the root folder and launch `yarn run dev`. This will launch `ts` in watch mode.

- Get into the `example` folder again and type `yarn run start`.

- You can now open Android Studio or XCode directly from `example/android` or `example/ios` respectively, build

and run the app.

## Tests & Verifications

Currently we use `eslint` with `prettier` for linting and formatting the code.

We still don't have a test suite (it's a WIP).

All of these are run on CircleCI for all opened pull requests, but you should use them locally when making changes.

- `yarn test`: Run all tests and validations.

- `yarn lint`: Run `eslint`.

- `yarn test:ts`: Checks TypeScript

- `yarn lint --fix`: Run `eslint` and automatically fix issues. This is useful for correcting code formatting.

## Sending a pull request

When you're sending a pull request:

- Prefer small pull requests focused on one change.

- Verify that all tests and validations are passing.

- Follow the pull request template when opening a pull request.

## Commit message convention

We prefix our commit messages with one of the following to signify the kind of change:

- \*\*build\*\*: Changes that affect the build system or external dependencies.

- \*\*ci\*\*, \*\*chore\*\*: Changes to our CI configuration files and scripts.

- \*\*docs\*\*: Documentation only changes.

- \*\*feat\*\*: A new feature.

- \*\*fix\*\*: A bug fix.

- \*\*perf\*\*: A code change that improves performance.

- \*\*refactor\*\*: A code change that neither fixes a bug nor adds a feature.

- \*\*style\*\*: Changes that do not affect the meaning of the code.

- \*\*test\*\*: Adding missing tests or correcting existing tests.

## Release process

We use [Semantic Release](http://semantic-release.org) to automatically release new versions of the library when changes are merged into master. Using the commit message convention described above, it will detect if we need to release a patch, minor, or major version of the library.

## Reporting issues

You can report issues on our [bug tracker](https://github.com/react-native-community/react-native-modal/issues). Please search for existing issues and follow the issue template when opening an issue.

## License

By contributing to React Native Modal, you agree that your contributions will be licensed under the \*\*MIT\*\* license.