# Chai Contribution Guidelines

We like to encourage you to contribute to the Chai.js repository. This should be as easy as possible for you but there are a few things to consider when contributing. The following guidelines for contribution should be followed if you want to submit a pull request or open an issue.

Following these guidelines helps to communicate that you respect the time of the developers managing and developing this open source project. In return, they should reciprocate that respect in addressing your issue or assessing patches and features.

#### Table of Contents

- [TLDR;](#tldr)

- [Contributing](#contributing)

- [Bug Reports](#bugs)

- [Feature Requests](#features)

- [Pull Requests](#pull-requests)

- [Releasing](#releasing)

- [Support](#support)

- [Resources](#resources)

- [Core Contributors](#contributors)

<a name="tldr"></a>

## TLDR;

- Creating an Issue or Pull Request requires a [GitHub](http://github.com) account.

- Issue reports should be \*\*clear\*\*, \*\*concise\*\* and \*\*reproducible\*\*. Check to see if your issue has already been resolved in the [master]() branch or already reported in Chai's [GitHub Issue Tracker](https://github.com/chaijs/chai/issues).

- Pull Requests must adhere to strict [coding style guidelines](https://github.com/chaijs/chai/wiki/Chai-Coding-Style-Guide).

- In general, avoid submitting PRs for new Assertions without asking core contributors first. More than likely it would be better implemented as a plugin.

- Additional support is available via the [Google Group](http://groups.google.com/group/chaijs) or on irc.freenode.net#chaijs.

- \*\*IMPORTANT\*\*: By submitting a patch, you agree to allow the project owner to license your work under the same license as that used by the project.

<a name="contributing"></a>

## Contributing

The issue tracker is the preferred channel for [bug reports](#bugs),

[feature requests](#features) and [submitting pull

requests](#pull-requests), but please respect the following restrictions:

\* Please \*\*do not\*\* use the issue tracker for personal support requests (use

[Google Group](https://groups.google.com/forum/#!forum/chaijs) or IRC).

\* Please \*\*do not\*\* derail or troll issues. Keep the discussion on topic and

respect the opinions of others

<a name="bugs"></a>

### Bug Reports

A bug is a \*\*demonstrable problem\*\* that is caused by the code in the repository.

Guidelines for bug reports:

1. \*\*Use the GitHub issue search\*\* &mdash; check if the issue has already been reported.

2. \*\*Check if the issue has been fixed\*\* &mdash; try to reproduce it using the latest `master` or development branch in the repository.

3. \*\*Isolate the problem\*\* &mdash; create a test case to demonstrate your issue. Provide either a repo, gist, or code sample to demonstrate you problem.

A good bug report shouldn't leave others needing to chase you up for more information. Please try to be as detailed as possible in your report. What is your environment? What steps will reproduce the issue? What browser(s) and/or Node.js versions experience the problem? What would you expect to be the outcome? All these details will help people to fix any potential bugs.

Example:

> Short and descriptive example bug report title

>

> A summary of the issue and the browser/OS environment in which it occurs. If suitable, include the steps required to reproduce the bug.

>

> 1. This is the first step

> 2. This is the second step

> 3. Further steps, etc.

>

> `<url>` - a link to the reduced test case OR

> ```js

> expect(a).to.equal('a');

> // code sample

> ```

>

> Any other information you want to share that is relevant to the issue being reported. This might include the lines of code that you have identified as causing the bug, and potential solutions (and your opinions on their merits).

<a name="features"></a>

### Feature Requests

Feature requests are welcome. But take a moment to find out whether your idea fits with the scope and aims of the project. It's up to \*you\* to make a strong case to convince the project's developers of the merits of this feature. Please provide as much detail and context as possible.

Furthermore, since Chai.js has a [robust plugin API](http://chaijs.com/guide/plugins/), we encourage you to publish \*\*new Assertions\*\* as plugins. If your feature is an enhancement to an \*\*existing Assertion\*\*, please propose your changes as an issue prior to opening a pull request. If the core Chai.js contributors feel your plugin would be better suited as a core assertion, they will invite you to open a PR in [chaijs/chai](https://github.com/chaijs/chai).

<a name="pull-requests"></a>

### Pull Requests

- PRs for new core-assertions are advised against.

- PRs for core-assertion bug fixes are always welcome.

- PRs for enhancing the interfaces are always welcome.

- PRs that increase test coverage are always welcome.

- PRs are scrutinized for coding-style.

Good pull requests - patches, improvements, new features - are a fantastic help. They should remain focused in scope and avoid containing unrelated commits.

\*\*Please ask first\*\* before embarking on any significant pull request (e.g. implementing features, refactoring code), otherwise you risk spending a lot of time working on something that the project's developers might not want to merge into the project.

Please adhere to the coding conventions used throughout a project (indentation, accurate comments, etc.) and any other requirements (such as test coverage). Please review the [Chai.js Coding Style Guide](https://github.com/chaijs/chai/wiki/Chai-Coding-Style-Guide).

Follow this process if you'd like your work considered for inclusion in the project:

1. [Fork](http://help.github.com/fork-a-repo/) the project, clone your fork, and configure the remotes:

```bash

# Clone your fork of the repo into the current directory

git clone https://github.com/<your-username>/<repo-name>

# Navigate to the newly cloned directory

cd <repo-name>

# Assign the original repo to a remote called "upstream"

git remote add upstream https://github.com/<upstream-owner>/<repo-name>

```

2. If you cloned a while ago, get the latest changes from upstream:

```bash

git checkout <dev-branch>

git pull upstream <dev-branch>

```

3. Create a new topic branch (off the main project development branch) to contain your feature, change, or fix:

```bash

git checkout -b <topic-branch-name>

```

4. Commit your changes in logical chunks. Use Git's [interactive rebase](https://help.github.com/articles/interactive-rebase) feature to tidy up your commits before making them public.

5. Run you code to make sure it works. If you're still having problems please try to run `make clean` and then test your code again.

```bash

npm test

# when finished running tests...

git checkout chai.js

```

6. Locally merge (or rebase) the upstream development branch into your topic branch:

```bash

git pull [--rebase] upstream <dev-branch>

```

7. Push your topic branch up to your fork:

```bash

git push origin <topic-branch-name>

```

8. [Open a Pull Request](https://help.github.com/articles/using-pull-requests/) with a clear title and description.

\*\*IMPORTANT\*\*: By submitting a patch, you agree to allow the project owner to license your work under the same license as that used by the project.

<a name="releasing"></a>

## Releasing

Releases can be \*\*prepared\*\* by anyone with access to the code.

Simply run `make release-major`, `make release-minor`, or `make-release-patch`

and it will automatically do the following:

- Build chai.js

- Bump the version numbers across the project

- Make a commit within git

All you need to do is push the commit up and make a pull request, one of the core contributors will merge it and publish a release.

### Publishing a Release

Anyone who is a core contributor (see the [Core Contributors Heading in the Readme](https://github.com/chaijs/chai#core-contributors)) can publish a release:

1. Go to the [Releases page on Github](https://github.com/chaijs/chai/releases)

2. Hit "Draft a new release" (if you can't see this, you're not a core contributor!)

3. Write human-friendly Release Notes based on changelog.

- The release title is "x.x.x / YYYY-MM-DD" (where x.x.x is the version number)

- If breaking changes, write migration tutorial(s) and reasoning.

- Callouts for community contributions (PRs) with links to PR and contributing user.

- Callouts for other fixes made by core contributors with links to issue.

4. Hit "Save Draft" and get other core contributors to check your work, or alternatively hit "Publish release"

5. That's it!

<a name="support"></a>

## Support

<a name="resources"></a>

### Resources

For most of the documentation you are going to want to visit [ChaiJS.com](http://chaijs.com).

- [Getting Started Guide](http://chaijs.com/guide/)

- [API Reference](http://chaijs.com/api/)

- [Plugins](http://chaijs.com/plugins/)

Alternatively, the [wiki](https://github.com/chaijs/chai/wiki) might be what you are looking for.

- [Chai Coding Style Guide](https://github.com/chaijs/chai/wiki/Chai-Coding-Style-Guide)

- [Third-party Resources](https://github.com/chaijs/chai/wiki/Third-Party-Resources)

Or finally, you may find a core-contributor or like-minded developer in any of our support channels.

- IRC: irc.freenode.org #chaijs

- [Mailing List / Google Group](https://groups.google.com/forum/#!forum/chaijs)

<a name="contributors"></a>

### Core Contributors

Feel free to reach out to any of the core-contributors with you questions or concerns. We will do our best to respond in a timely manner.

- Jake Luer

- GH: [@logicalparadox](https://github.com/logicalparadox)

- TW: [@jakeluer](http://twitter.com/jakeluer)

- IRC: logicalparadox

- Veselin Todorov

- GH: [@vesln](https://github.com/vesln/)

- TW: [@vesln](http://twitter.com/vesln)

- IRC: vesln

- Keith Cirkel

- GH: [@keithamus](https://github.com/keithamus)

- TW: [@keithamus](http://twitter.com/keithamus)

- IRC: keithamus

- Lucas Fernandes da Costa

- GH: [@lucasfcosta](https://github.com/lucasfcosta)

- TW: [@lfernandescosta](https://twitter.com/lfernandescosta)

- IRC: lucasfcosta