# Contributing to Streamlink

Want to get involved? Thanks! There are plenty of ways to help!

## Reporting issues

A bug is a \*demonstrable problem\* that is caused by the \*\*latest version\*\* of the code in the repository. Good bug reports are extremely helpful - thank you!

Please read the following guidelines before you [report an issue][issues]:

1. \*\*See if the issue is already known\*\* — check the list of [known issues][known-issues].

2. \*\*Use the GitHub issue search\*\* — check if the issue has already been reported. If it has been, please comment on the existing issue.

3. \*\*Check if the issue has been fixed\*\* — the latest `master` or development branch may already contain a fix.

4. \*\*Isolate the demonstrable problem\*\* — make sure that the code in the project's repository is \*definitely\* responsible for the issue.

5. \*\*Format your issue report\*\* — [well formatted texts][mastering-markdown] will make it much easier for developers to understand your report.

Please try to be as detailed as possible in your report too. What is your environment? What steps will reproduce the issue? What would you expect the outcome to be? All these details will help people to assess and fix any potential bugs. The [issue template][issue-template] will aid you in structuring your report when submitting a new issue. If your report includes log output, please provide it as text in multi-line code blocks if possible for better readability. Thank you!

## 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.

## Plugin requests

Plugin submissions and requests are a great way to improve Streamlink. Requests should be as detailed as possible and dedicated to only a single streaming service. Information about the service as well as explicit URLs for live streams are required, as are details regarding the website, what it is, who runs it, how it adds to Streamlink, etc. This allows for easier plugin review and implementation. We reserve the right to refuse to implement, develop, or update any plugin. In addition we may remove existing plugins at our own discretion.

Plugins which fall under the following categories will not be implemented or considered and the request will be closed:

1. Any kind of streaming service that uses DRM protection

2. Sites which are hosting stolen content as their main source of content

3. Sites which are primarily rehosting content that is available from a legitimate source (TV shows, sports, news, etc.)

4. Sites which require any sort of cable login or subscription

5. NSFW sites of a pornographic nature (cam sites, porn sites, etc.)

6. Sites which are mostly hosting VODs (stolen or legitimate), or require large changes to existing code to watch VODs

7. Sites where the maintainer has requested we not add their site to Streamlink

8. Sites which are unmaintained.

## Pull requests

Good pull requests - patches, improvements, and 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, porting to a different language), 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, white space, accurate comments, etc.) and any other requirements (such as test coverage).

Adhering to the following process is the best way to get your work included in the project:

1. [Fork][howto-fork] the project, clone your fork, and configure the remotes:

```bash

# Clone your fork of the repo into the current directory

git clone git@github.com:<YOUR-USERNAME>/streamlink.git

# Navigate to the newly cloned directory

cd streamlink

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

git remote add upstream https://github.com/streamlink/streamlink.git

```

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

```bash

git checkout master

git pull upstream master

```

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

```bash

git checkout -b <TOPIC-BRANCH-NAME>

```

4. Commit your changes in logical chunks. Please adhere to these [git commit message guidelines][howto-format-commits] or your code is unlikely be merged into the project. Use git's [interactive rebase][howto-rebase] feature to tidy up your commits before making them public.

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

```bash

git pull [--rebase] upstream master

```

6. Push your topic branch up to your fork:

```bash

git push origin <TOPIC-BRANCH-NAME>

```

7. [Open a Pull Request][howto-open-pull-requests] with a clear title and description.

\*\*IMPORTANT\*\*: By submitting a patch, you agree to allow the project owners to license your work

under the terms of the [BSD 2-clause license][license].

## Acknowledgements

This contributing guide has been adapted from [HTML5 boilerplate's guide][ref-h5bp].

[issues]: https://github.com/streamlink/streamlink/issues

[known-issues]: https://github.com/streamlink/streamlink/blob/master/KNOWN\_ISSUES.md

[issue-template]: https://github.com/streamlink/streamlink/blob/master/ISSUE\_TEMPLATE.md

[mastering-markdown]: https://guides.github.com/features/mastering-markdown

[howto-fork]: https://help.github.com/articles/fork-a-repo

[howto-rebase]: https://help.github.com/articles/interactive-rebase

[howto-format-commits]: http://tbaggery.com/2008/04/19/a-note-about-git-commit-messages.html

[howto-open-pull-requests]: https://help.github.com/articles/using-pull-requests

[Git]: https://git-scm.com

[license]: https://github.com/streamlink/streamlink/blob/master/LICENSE

[ref-h5bp]: https://github.com/h5bp/html5-boilerplate/blob/master/CONTRIBUTING.md