# Contributing to webpack-dev-server

Do you use webpack-dev-server and want to help us out? Thanks!

Please review this document before contributing.

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 show respect in addressing your issue or assessing patches and features.

## Core Ideas

- There are hooks to add your own features, so we should not add less-common features.

- The workflow should be to start webpack-dev-server as a separate process, next to the "normal" server and to request the script from this server or to proxy from dev-server to "normal" server (because webpack blocks the event queue too much while compiling which can affect "normal" server).

- A user should not try to implement stuff that accesses the webpack filesystem. This lead to bugs (the middleware does it while blocking requests until the compilation has finished, the blocking is important).

- It should be a development only tool. Compiling in production is bad, one should precompile and deliver the compiled assets.

- Processing options and stats display is delegated to webpack, so webpack-dev-server/middleware should not do much with it. This also helps us to keep up-to-date with webpack updates.

- The communication library (`SockJS`) should not be exposed to the user.

## Submitting a Pull Request

Good pull requests, such as patches, improvements, and new features, are a fantastic help. They should remain focused in scope and not contain unrelated commits.

It is advised to first create an issue (if there is not one already) before making a pull request. This way the maintainers can first discuss with you if they agree and it also helps with providing some context.

Run the relevant [examples](https://github.com/webpack/webpack-dev-server/tree/master/examples) to see if all functionality still works. When introducing new functionality, also add an example. This helps the maintainers to understand it and check if it still works.

## Setting Up a Local Copy

1. Clone the repo with `git clone https://github.com/webpack/webpack-dev-server`.

2. Run `npm install` in the root `webpack-dev-server` folder.

Once it is done, you can modify any file locally. In the `examples/` directory you'll find a lot of examples with instructions on how to run it. This can be very handy when testing if your code works.

If you are modifying a file in the `client/` directory, be sure to run `npm run prepublish` after it. This will recompile the files.

## Testing a Pull Request

Pull requests often need some real-world testing.

1. In your `package.json`, change the line with `webpack-dev-server` to:

```json

"webpack-dev-server": "<PATH>"

```

`<PATH>`:

- `github:webpack/webpack-dev-server#pull/<ID>/head`

where `<ID>` is the ID of the pull request.

- `file:../path/to/local/webapck-dev-server/fork` is the path to your local repo, just make sure you hit the correct path

2. Run `npm install`.

3. Go to the `webpack-dev-server` module (`cd node\_modules/webpack-dev-server`), and run `npm run prepublish`.

The pull request is now ready to be tested.

## Contributor License Agreement

When submitting your contribution, a CLA (Contributor License Agreement) bot will come by to verify that you signed the [CLA](https://cla.js.foundation/webpack/webpack-dev-server). If it is your first time, it will link you to the right place to sign it. However, if you have committed your contributions using an email that is not the same as your email used on GitHub, the CLA bot can't accept your contribution.

Run `git config user.email` to see your Git email, and verify it with [your GitHub email](https://github.com/settings/emails).

---

\_Many thanks to [create-react-app](https://github.com/facebookincubator/create-react-app/blob/master/CONTRIBUTING.md) for the inspiration with this contributing guide\_