# Contributing to Vue CLI

## Workflow

The Git workflow used in this project is largely inspired by [Gitflow workflow](https://www.atlassian.com/git/tutorials/comparing-workflows/gitflow-workflow).

There are two main branches: `master` and `next`, corresponding to the npm `dist-tag`s with the same names.

The documentation website for the current CLI version <https://cli.vuejs.org> is deployed from the `master` branch, while documentation for new features <https://next.cli.vuejs.org/> is deployed from `next` branch.

When sending documentation pull requests, please fork your branches from these two branches.

The development branch is `dev`.

And there are several version branches for archiving old versions of Vue CLI, such as `v2`, `v3`.

Pull requests that touches the code should be forked from `dev`, unless it's only targeting an old version.

## Development Setup

This project uses a monorepo setup that requires using [Yarn](https://yarnpkg.com) because it relies on [Yarn workspaces](https://yarnpkg.com/blog/2017/08/02/introducing-workspaces/).

``` sh

# install dependencies

yarn

# link `vue` executable

# if you have the old vue-cli installed globally, you may

# need to uninstall it first.

cd packages/@vue/cli

yarn link

# create test projects in /packages/test

cd -

cd packages/test

vue create test-app

cd test-app

yarn serve

```

### Testing Tips

The full test suite is rather slow, because it has a number of e2e tests that perform full webpack builds of actual projects. To narrow down the tests needed to run during development, you can pass the test script a regex to match test filenames:

``` sh

yarn test <filenameRegex>

```

Note the regex matches against full paths relative to the project root, so for example if you want to test all the prompt modules in `packages/@vue/cli/lib/promptModules`, you can simply run:

``` sh

yarn test promptModules

```

Alternatively, you can run the tests inside specific packages with the `-p` flag:

``` sh

yarn test -p cli,cli-service

```

If the package is a plugin, you can omit the `cli-plugin-` prefix:

``` sh

yarn test -p typescript

```

You can also pass `--watch` to run tests in watch mode.

Note that `jest --onlyChanged` isn't always accurate because some tests spawn child processes.

### Plugin Development

See [dedicated section in docs](https://cli.vuejs.org/dev-guide/plugin-dev.html).