# How to Contribute #

We're always looking for people to help make Lidarr even better, there are a number of ways to contribute.

## Documentation ##

Setup guides, FAQ, the more information we have on the wiki the better.

## Development ##

### Tools required ###

- Visual Studio 2017 or higher (https://www.visualstudio.com/vs/). The community version is free and works (https://www.visualstudio.com/downloads/).

- HTML/Javascript editor of choice (VS Code/Sublime Text/Webstorm/Atom/etc)

- [Git](https://git-scm.com/downloads)

- [NodeJS](https://nodejs.org/en/download/) (Node 8.X.X or higher)

- [Yarn](https://yarnpkg.com/)

- .NET 4.6.2 or Mono equivalent.

### Getting started ###

1. Fork Lidarr

2. Clone the repository into your development machine. [\*info\*](https://help.github.com/articles/working-with-repositories)

3. Grab the submodules `git submodule init && git submodule update`

4. Install the required Node Packages `yarn install`

5. Start gulp to monitor your dev environment for any changes that need post processing using `yarn start` command.

6. Build the project in Visual Studio, Setting startup project to `NZBDrone.Console`

7. Debug the project in Visual Studio

8. Open http://localhost:8686

### Contributing Code ###

- If you're adding a new, already requested feature, please comment on [Github Issues](https://github.com/lidarr/Lidarr/issues "Github Issues") so work is not duplicated (If you want to add something not already on there, please talk to us first)

- Rebase from Lidarr's develop branch, don't merge

- Make meaningful commits, or squash them

- Feel free to make a pull request before work is complete, this will let us see where its at and make comments/suggest improvements

- Reach out to us on the discord if you have any questions

- Add tests (unit/integration)

- Commit with \*nix line endings for consistency (We checkout Windows and commit \*nix)

- One feature/bug fix per pull request to keep things clean and easy to understand

- Use 4 spaces instead of tabs, this is the default for VS 2017 and WebStorm (to my knowledge)

### Pull Requesting ###

- Only make pull requests to develop, never master, if you make a PR to master we'll comment on it and close it

- You're probably going to get some comments or questions from us, they will be to ensure consistency and maintainability

- We'll try to respond to pull requests as soon as possible, if its been a day or two, please reach out to us, we may have missed it

- Each PR should come from its own [feature branch](http://martinfowler.com/bliki/FeatureBranch.html) not develop in your fork, it should have a meaningful branch name (what is being added/fixed)

- new-feature (Good)

- fix-bug (Good)

- patch (Bad)

- develop (Bad)

If you have any questions about any of this, please let us know.