

This page is dedicated to help you, the new Tiki developer, get started in Tiki Development.

About

Please take the time to read this page before you start. We hope it will save you time in the long run. There are a lot of references here - some are basic, some advanced. You may need to read some or all of the references, but As you learn, we invite you to come back and update these pages - share what you have learned.

Getting started: The Cheat Sheet

1. Join the tikiwiki-devel mailing lists at https://sourceforge.net/mail/?group_id=64258 and ask lots of questions.
2. Join the [Tiki community chat](#) and ask lots of questions.
3. Use Tiki and understand how it works before you start trying to change it. Set up a site, use and administer it on an everyday basis until you can fix the problems you encounter.
4. Set up your tiki in development mode using one or more of the Git branches (there are several). The bleeding edge is *master*.

```
git clone --branch master https://gitlab.com/tikiwiki/tiki.git mynewfolder cd mynewfolder sh
setup.sh #Your choice [b]? b php console.php dev:configure
```

More details: [Installation](#) and [Setup a Development Environment](#).

5. Run main tests locally
...so you don't have to wait for CI

```
php phpunit php vendor_bundled/vendor/squizlabs/php_codesniffer/bin/phpcs --cache
```

6. Fix the errors
Some tools are useful, such as:

```
php console.php dev:fixstyle #Very useful, but only fixes files not already committed
```

More details:

1. [Tiki Unit Testing](#)
 2. [PHP Code Sniffer](#)
7. Commit and create a merge request
This is done through https://gitlab.com/tikiwiki/tiki/-/merge_requests/new
More details:
 1. [How to commit](#).
 2. [Where to commit](#).

What else your should read...

1. Read the [3 Rules](#).
2. If this is the first open-source project you have worked on, understand that working on an open-source project is **NOT** like any other development project you have ever done. You will be dealing with a large number of people who are scattered all over this beautiful world of ours. Many of our friends learned English as a second (or third) language. Misunderstandings **WILL** happen! Assume positive intent and practice forgiveness. All of this will require an added effort on your part. Please

read [KinderCode](#) often. A good place to learn about the differences of open source development is Eric Raymond's [Cathedral and the Bazaar](#) [↗](#).

3. **Before** you DO anything - ask for help on the Development list or on the community chat. It doesn't make a lot of sense for you to spend a lot of time trying to accomplish something the **Hard Way** when a few questions could save you a lot of effort. Again - ask questions and use the expertise that is available.
4. If you have an idea for a new feature, suggest it on the dev list or #tikiwiki. Get some feedback and opinions on the need for the feature and ideas on how to implement it. Some in the Tiki community have been here for 2-3 years and know the history. As Newton before you, take an opportunity to see further by *standing on the shoulders of giants*.
5. If you need help with something - **ASK!** Most developers are more than happy to assist when they can - but we need to know when (and where) the help is needed. This is a **very** friendly group who believe in helping each other!
6. Read the [Hello World](#) introduction in order to learn the basic structure of the Tiki environment. This article starts with a basic "hello world" page and goes on to cover permissions, menus, queries, wiki parsing, themes and more.
7. [How to get commit access](#)
8. Tiki is licensed under [the LGPL license](#) [↗](#). Take the time to understand the meaning of this license before you commit any code. By submitting to the project you are committing to that license model. Further, any code you copy from another open source project must be compatible with LGPL.
9. Finally: That nasty word **Documentation**. While nobody expects a developer to spend his time writing the documentation for the manual, some documentation is necessary. Writing a simple proposal in a Wiki page on dev.tiki.org stating what you want to accomplish and how you plan to do it does several things. First, it will help you to clarify your thoughts. Second, it should provide you and others with some specific goals. The most important thing is that it will provide others with a place where they can comment on your ideas. Documenting the source code is also **important!** Comments allow the unskilled novice to become better.

Other development references:

- [DevTips](#)
- [Git Usage](#)
- [Introduction to Tiki Code Layout](#)

Aliases

[Get Started](#) | [GetStarted](#)