Documentation Page   menu

Documentation Page

The official docs for Documentation Page. Talk about dogfeeding 🤗

Getting Started

There is a high chance that your documentation already works properly! Try it out by searching your repos in our homepage:

https://documentation.page/

You can organize your documentation broadly in three different ways:

- Single **readme.md** file. Ideal for smaller projects, you can put all the docs in this file and call it a day. See example.

- Folder **documentation** with multiple markdown files. These will all be concatenated in alphabetic order to generate the full documentation. (TODO: examples).

- Custom **documentation.page.json** configuration if you want a different personalized docs organization. See example config with output website.

Convention

This is the most powerful bit, since we will crawl your repository looking for some specific helpful bits and use that to generate the website. When possible, we recommend following the convention first, and otherwise adding some custom configuration.

The data that is parsed automatically is similar to the configuration available:

key	description	source
name	very short identifier for the project	Github API's repo `name`, cleaned up
title	longer identifier for the project	Same as `name`
description	a paragraph summarizing the project	Github API's repo `description`

key	description	source
logo	absolute path pointing to your project logo	Any file found called `favicon.png` or `logo.png`
menu	the top-right menu links for custom domains	Configuration **required** for custom domains
pages	list the files/folders to crawl	`documentation` if found or `readme.md` otherwise

Configuration

These are the documentation options right now:

key	description	example
name	very short identifier for the project	Statux
title	longer identifier for the project	Statux • The easy React state management library
description	a paragraph summarizing the project	A straightforward way of dealing with your ...
logo	absolute path pointing to your project logo	https://example.com/mylogo.png
menu	the top-right menu links for custom domains	{ "Donate": "...", "Github": "...", ... }
pages	list the files/folders to crawl	["readme.md", "documentation", "examples", ...]

They are the keys of a JSON file, which can be either of these:

- `documentation.page.json`: a file on the root of the project
- `documentation/page.json`: a file called `page.json` inside a folder called `documentation`
- [not yet] the key `documentation` inside your `package.json`

name

A very short name to identify the project. It must be the name of the project and nothing else, and so it excludes the tag line, icon, etc. It will be visible in:

- Tab title
- Navbar title (Pro)

title

Longer identifier for the project, possibly stating the tagline. It will be visible in:

- Google search main title
- Share card preview main title

description

A paragraph summarizing the project funcionality and competitive advantages. This appears in:

- Google search description
- Share card preview description

logo

An absolute path to the image associated with the logo of your project. It can only be in `.png` right now. It will show on:

- Favicon
- Navbar before the project name

menu

the top-right menu links for custom domains. It is an object with the labels (text) as the keys, and the links as the URLs. Example:

```
{
  "menu": {
    "Issues": "https://github.com/franciscop/react-test/issues",
    "Contribute": "https://github.com/franciscop/react-test/blob/mast
    "Donate": "https://www.paypal.me/franciscopresencia/19",
    "Github": "https://github.com/franciscop/react-test"
```

```
    }
  }
```

pages

Where the data should be searched. Defaults to `heroreadme.md`. It IS case insensitive.
Example:

```
{
  "pages": ["readme.md", "src/methods", "src/jest-matchers", "src/exa
}
```

Guides

Moving OUT of Documentation Page

I created Documentation Page to be a place where developers can easily semi-automate their docs websites, not to lock them in. So if at some point, for some reason you decide to move out into e.g. Github Pages, please feel free to do so! There are some disadvantages but these might not affect you.

The easiest and probably best way of making your Github Pages is having a single markdown file with all of the docs. Then you can either have a script that converts that into a full HTML page locally, or using a Jekyll template that converts it into an HTML page on Github pages.

[TODO]

How it works

Reading the repo

The repository specified must be public so that it can be read by Documentation Page (contact us if you need a private one). Once you specify your username+repo, Documentation Page gets to work. It'll parse different files to find the best way to build the documentation.

You can organize your documentation broadly in three different ways:

- Single **readme.md** file. Ideal for smaller projects, you can put all the docs in this file and call it a day. See example.

- Folder **documentation** with multiple markdown files. These will all be concatenated in alphabetic order to generate the full documentation. For this, it is recommended to add indexes like `1. Welcome`, `2. Getting Started` to your files. (TODO: examples).

- Custom **documentation.page.json** configuration if you want a different personalized docs organization. See example config with output website.

The way Documentation Page works is, we merge all of those files into a single large file.

Autogenerated Navigation

The navigation is autogenerated from the merged file containing all of your documentation. It's parsed, put together, and then sorted from h1 to h4. If there are multiple h1s, we drop all the headers a level except the first h1.

Full-text Search

Compare with

Github Pages

Documentation Page

✦ About us

✦ Getting Started

✦ Documentation

✦ Pricing

Examples

✦ Picnic CSS

✦ Umbrella JS

✦ React Test

✦ Statux

This is an early beta website, for more info please contact or follow me:

✦ francisco.io

✦ twitter.com/fpresencia