gpbl Follow
I'm a passionate React.js and Swift developer. Writing from everywhere in the world .
Nov 14, 2015 · 4 min read

Using Gitbook to document an open source project

Gitbook is a great tool for creating digital books. It also works very well for publishing documentation of open source projects.

Here I'll describe how to publish on Github Pages the documentation book for a **npm package** hosted on Github.

What I like about this solution are the docs written in markdown and saved in the project's repository. They can easily published with a single npm script. Everyone can contribute to them with a simple pull request.

I use this technique for my react-day-picker project, however the idea has been heavily borrowed from the redux repo. If you are not familiar with Gitbook, I suggest you to read the documentation first.

Prepare the book

The repository's main **README.md** is going to be the book's introduction.

Create a **book.json** in the project’s root. This is used by Gitbook for its settings. In our case, we want the book’s table of content in *docs/SUMMARY.md*:

```
{
  "gitbook": "2.5.2",
  "structure": {
    "summary": "docs/SUMMARY.md"
  }
}
```

(For now, until this issue is solved, make sure you use gitbook 2.5.2)

Now create a simple **SUMMARY.md** and place it in the **docs** directory. This is where you write out the tree of the book. For example, use this content:

```
# Table of content

* [Getting Started](docs/getting-started.md)
* [API Reference](docs/api-reference.md)
```

Now create the two files **getting-started.md** and **api-reference.md**, put some content and save them in the **docs** directory.

You can already start writing the Gitbook! Keep in mind:

- the project's *README.md* is the book's introduction
- *docs/SUMMARY.md* contains the table of content (ToC) — you have to add there the book's chapters and articles
- save all the markdown files linked from the ToC in the */docs* directory or its subdirectories

Previewing the book

Install the gitbook-cli and add it to the development dependencies:

```
npm install gitbook-cli --save-dev
```

Add the **following scripts** to *package.json*:

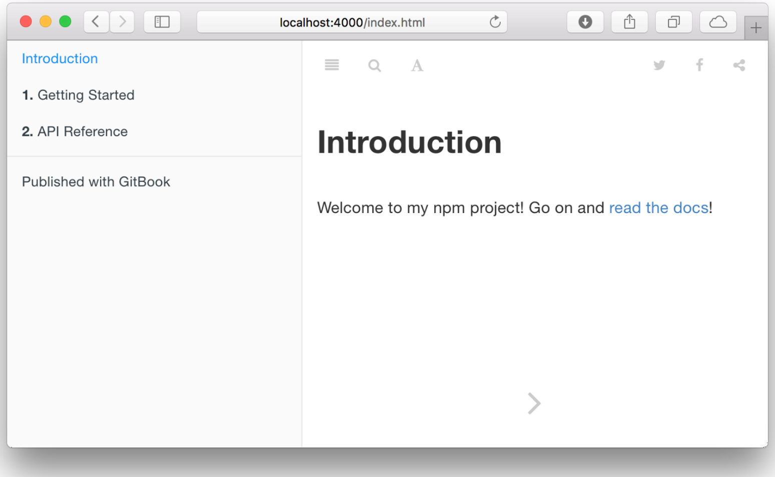
```
"scripts": {
   "docs:prepare": "gitbook install",
   "docs:watch": "npm run docs:prepare && gitbook serve"
}
```

docs:prepare will set-up gitbook (such as dependencies, plugins etc.) and *docs:watch* will start a server showing the book.

Now, run the watch task:

```
npm run docs:watch
```

and open localhost:4000:



localhost:4000/index.html

Introduction

1. Getting Started

2. API Reference

Published with GitBook

Introduction

Welcome to my npm project! Go on and read the docs!

Here you go, the Gitbook! As you change the content of the book, the browser will reload it automatically.

Publish the book on Github Pages

If you haven’t done it yet, create manually the **gh-pages branch** following this guide on github.com. Content pushed on this branch will be published on *http://[username].github.io/[repo]*.

Switch again to the *master* branch, then add **another script** to *package.json*:

```
"scripts": {
  "docs:build": "npm run docs:prepare && rm -rf _book &&
gitbook build"
}
```

Running *npm docs:build* will create a **_book** directory with the Gitbook website we want to publish: indeed, if you open *_book/index.html* you will see the static website generated by Gitbook. (remember to add *_book* to your *.gitignore*)

We want to create now a **npm script** that publishes the content of *_book* to the *gh-pages* branch. This part is a bit more complex, since we have to init a new repository and push it to the branch. Anyway, the sequence of the commands to run is the following:

```
$ npm run docs:build
$ cd _book
$ git init
$ git commit --allow-empty -m 'Update docs'
$ git checkout -b gh-pages
$ git add .
$ git commit -am 'Update docs'
$ git push git@github.com:<username>/<repo> gh-pages --force
```

Put all in one line, add the following **docs:publish** script to the *package.json* scripts section:

```
"docs:publish": "npm run docs:build && cd _book && git init
&& git commit --allow-empty -m 'Update docs' && git checkout
-b gh-pages && git add . && git commit -am 'Update docs' &&
git push git@github.com:<username>/<repo> gh-pages --force"
```

Rembember to change <*username*> and <*repo*> with your Github username and the name of the repo!

Now to publish the book to the github page, you have just to enter:

```
npm run docs:publish
```

Wait some minutes, and visit *http://[username].github.io/[repo]*. If everything is gone right, you should see the books published there.

Now, with Gitbook and Github Pages, documenting your open source project is really fun! Just change the markdown and launch *npm run docs:publish*.

PS. You can see the source files described above on this gitbook-demo repo.

Some nice additions

- Gitbook support a variety of plugins. I've found useful to replace the original code highlighting plugin with another one based on Prism. The book.json file should disable the *highlight* plugin:

```
"plugins": ["prism", "-highlight"]
```

- You can tweak the Gitbook's CSS with your own. Create a new .css file and add a reference to it in *book.json*, for example:

```
"styles": {
  "website": "docs/styles/website.css"
}
```

- Add some trendy badges from http://shields.io :)