
sphinxcontrib-editable

Release 0.1.0

Oct 08, 2019

Contents

1	Overview	3
2	Live demo	5
3	Install	7
4	Documentation	9
5	Source code	11
6	Contribute	13
7	Contents	15
7.1	sphinxcontrib-editable	15
7.2	Installation	16

In-line editable documentation

CHAPTER 1

Overview

I read a [conversation](#) which discusses the advantages of Google Docs over traditional markup documentation because it lets users provide corrections and feedback inline. One notable section:

it's a dynamic and collaborative experience. It's a living process. **It makes it trivially easy for anyone, with or without an account, with or without programming skills, to update the content and collaborate with me on it, in-context.** To pop in a correction for a wrong number (so many corrections so far!). To reword a phrase poorly constructed (thank you readers!). The experience is substantially distinct and better than a wiki as well, especially if you have non-technical readers. Comments and suggestions are treated differently than content, and integrated with a notification system via email. There's no awkward syntax to learn. Even technical users have a difficult time keeping wikis up-to-date! I'm witnessing it daily, both in the OSS world and in the commercial sphere. Maintaining up-to-date docs is REALLY difficult for engineers. Here . . . well you just make the edit right there in front of your eyes as you notice something that needs a change and an email is automatically sent to the author.

These sound like important benefits of using Google Docs. But I want to write my docs in a markup language because it's much more flexible. So my ideal documentation system would have several features:

- Maintainers can write docs in a powerful markup language.
- Users can edit the docs in-line as they're reading, without needing to log in or know reStructuredText.
- Users can comment on the docs with points of clarification. ([sphinxcontrib-disqus](#) provides a comments section at the bottom of the page, but no in-line comments. This is still a missing piece.)

This library demos no-login editing capability for Sphinx docs.

CHAPTER 2

Live demo

Try out a [live demo](#) here.

CHAPTER 3

Install

Install this package:

```
pip install sphinxcontrib-editable
```

And in `docs/conf.py` add

```
extensions = ['sphinxcontrib.editable']
```


CHAPTER 4

Documentation

<https://sphinxcontrib-editable.readthedocs.io/>

CHAPTER 5

Source code

<https://github.com/metatooling/sphinxcontrib-editable>

CHAPTER 6

Contribute

Come to the issue tracker [on GitHub](#).

7.1 sphinxcontrib-editable

In-line editable documentation

7.1.1 Overview

I read a [conversation](#) which discusses the advantages of Google Docs over traditional markup documentation because it lets users provide corrections and feedback inline. One notable section:

it's a dynamic and collaborative experience. It's a living process. **It makes it trivially easy for anyone, with or without an account, with or without programming skills, to update the content and collaborate with me on it, in-context.** To pop in a correction for a wrong number (so many corrections so far!). To reword a phrase poorly constructed (thank you readers!). The experience is substantially distinct and better than a wiki as well, especially if you have non-technical readers. Comments and suggestions are treated differently than content, and integrated with a notification system via email. There's no awkward syntax to learn. Even technical users have a difficult time keeping wikis up-to-date! I'm witnessing it daily, both in the OSS world and in the commercial sphere. Maintaining up-to-date docs is REALLY difficult for engineers. Here ... well you just make the edit right there in front of your eyes as you notice something that needs a change and an email is automatically sent to the author.

These sound like important benefits of using Google Docs. But I want to write my docs in a markup language because it's much more flexible. So my ideal documentation system would have several features:

- Maintainers can write docs in a powerful markup language.
- Users can edit the docs in-line as they're reading, without needing to log in or know reStructuredText.
- Users can comment on the docs with points of clarification. ([sphinxcontrib-disqus](#) provides a comments section at the bottom of the page, but no in-line comments. This is still a missing piece.)

This library demos no-login editing capability for Sphinx docs.

7.1.2 Live demo

Try out a [live demo](#) here.

7.1.3 Install

Install this package:

```
pip install sphinxcontrib-editable
```

And in docs/conf.py add

```
extensions = ['sphinxcontrib.editable']
```

7.1.4 Documentation

<https://sphinxcontrib-editable.readthedocs.io/>

7.1.5 Source code

<https://github.com/metatooling/sphinxcontrib-editable>

7.1.6 Contribute

Come to the issue tracker [on GitHub](#).

7.2 Installation

At the command line:

```
pip install sphinxcontrib-editable
```

- genindex
- modindex
- search