sphinx-btn

Release 0.1.2

Pierrick Rambaud

Mar 14, 2023

CONTENTS

CHAPTER

ONE

QUICKSTART

This section contains basic information about **sphinx-btn** to get you started.

1.1 Installation

Use `pip` to install **sphinx-btn** in your environment:

```
pip install sphinx-btn
```

1.2 Extension setup

After installing **sphinx-btn**, add `sphinxcontrib.icon` and `sphinxcontrib.btn` to the list of extensions in your `conf.py` file:

```
extensions = [
    #[...]
    "sphinxcontrib.icon",
    "sphinxcontrib.btn"
]
```

Note: The **sphinx-btn** extention rely on **sphinx-icon** to provide access to the fontawesome 6.3.0 icons font/metadata.

1.3 Icon directive

You can now add buttons directly in your documentation:

```
I'm a :btn:`<fa-solid fa-folder> fa-folder` btn.
I'm a :btn:`<fa-solid fa-folder>` btn.
I'm a :btn:`fa-folder` btn.
```

I'm a fa-folder btn.

I'm a btn.

I'm a fa-folder btn.

Note: Support is provided for older version of Fontawesome. Documentation using `fas | far | fab` or `fa` will continue working. Be aware that the icon you want to use may changed name since then.

1.4 HTML output

In the HTML output, the CSS and JS from Fontawesome 6.3.0 are added to the output in the `<head>` tag.

```
<link rel="stylesheet" type="text/css" href="<webpath>/build/html/_font/fontawesome/
↪css/all.min.css">
<!-- -->
<script src="<webpath>/build/html/_font/fontawesome/css/all.min.js">
```

Then for each btn role occurence an `<span>` of class `guilabel` tag will be used:

```
<span class="guilabel">
    <i class="fa-solid fa-folder"></i>
    <span style="margin-left: .5em;">fa-folder</span>
</span>
```

1.5 Latex output

For the latex output, the **sphinx-btn** extention need to use the webfonts provided by fontawesome. It will thus force the use of the XeLaTex builder to allow use of the fontspec and tcolorbox packages. Then 3 new font will be added to the preamble of the tex file as well as a `sphinxbtn` command:

```
\newfontfamily{\solid}{fa-solid-900.ttf}
\newfontfamily{\regular}{fa-regular-400.ttf}
\newfontfamily{\brands}{fa-brands-400.ttf}

\newtcbox{\sphinxbtn}[1][]{box align=base, nobeforeafter, size=small, boxsep=2pt, #1}
```

Then for each btn role occurence the following command will be used:

```
\sphinxbtn{{\solid\symbol{"F07B}} fa-folder}
```

where `solid` is the font style selected in the role and `F007` being the unicode of the selected icon.

CHAPTER

TWO

CONTRIBUTE

Thank you for your help improving **sphinx-btn**!

sphinx-btn uses nox to automate several development-related tasks. Currently, the project uses four automation processes (called sessions) in `noxfile.py`:

- `mypy`: to perform a mypy check on the lib;
- `test`: to run the test with pytest;
- `docs`: to build the documentation in the `build` folder;
- `lint`: to run the pre-commits in an isolated environment

Every nox session is run in its own virtual environment, and the dependencies are installed automatically.

To run a specific nox automation process, use the following command:

```
nox -s {{session name}}
```

For example: `nox -s test` or `nox -s docs`.

2.1 Workflow for contributing changes

We follow a typical GitHub workflow of:

- Create a personal fork of this repo
- Create a branch
- Open a pull request
- Fix findings of various linters and checks
- Work through code review

See the following sections for more details.

2.2 Clone the repository

First off, you'll need your own copy of **sphinx-btn** codebase. You can clone it for local development like so:

Fork the repository so you have your own copy on GitHub. See the GitHub forking guide for more information.

Then, clone the repository locally so that you have a local copy to work on:

```
git clone https://github.com/{{ YOUR USERNAME }}/btn
cd btn
```

Then install the development version of the extension:

```
pip install -e .[dev]
```

This will install the **sphinx-btn** library, together with two additional tools: - pre-commit for automatically enforcing code standards and quality checks before commits. - nox, for automating common development tasks.

Lastly, activate the pre-commit hooks by running:

```
pre-commit install
```

This will install the necessary dependencies to run pre-commit every time you make a commit with Git.

2.3 Contribute to the codebase

Any larger updates to the codebase should include tests and documentation. The tests are located in the `tests` folder, and the documentation is located in the `docs` folder.

To run the tests locally, use the following command:

```
nox -s test
```

See *below* for more information on how to update the documentation.

2.4 Contribute to the docs

The documentation is built using Sphinx and deployed to Read the Docs.

To build the documentation locally, use the following command:

```
nox -s docs
```

For each pull request, the documentation is built and deployed to make it easier to review the changes in the PR. To access the docs build from a PR, click on the "Read the Docs" preview in the CI/CD jobs.

The btn extension allows you to embed fontawesome 6.3.0 icons in guilabels as inline roles in a sphinx documentation.

```
I'm a :btn:`<fa-solid fa-folder> fa-folder` btn.
I'm a :btn:`<fa-solid fa-folder>` btn.
I'm a :btn:`fa-folder` btn.
```

I'm a fa-folder btn.

I'm a btn.

I'm a fa-folder btn.

CHAPTER

THREE

DOCUMENTATION CONTENTS

The documentation contains 2 sections:

Quickstart Lib instalation and usage of the video directive

Contribute Help us improve the Sphinx extention.