
github*reviewboard_syncDocumentation*

Release 0.1.0

Apptimize - Tim Martin

June 03, 2016

1	Installation	3
2	Usage	5
3	Contributing	7
3.1	Types of Contributions	7
3.2	Get Started!	8
3.3	Pull Request Guidelines	8
3.4	Tips	9
4	Credits	11
4.1	Development Lead	11
4.2	Contributors	11
5	History	13
5.1	0.1.6 (unreleased)	13
5.2	0.1.5 (2016-06-03)	13
5.3	0.1.4 (2016-05-09)	13
5.4	0.1.3 (2016-05-06)	13
5.5	0.1.2 (2016-05-05)	13
6	github-reviewboard-sync	15
6.1	Overview	15
6.2	Installation	15
6.3	Features	15
6.4	Credits	16
7	Indices and tables	17

Contents:

Installation

At the command line:

```
$ easy_install github-reviewboard-sync
```

Or, if you have virtualenvwrapper installed:

```
$ mkvirtualenv github-reviewboard-sync  
$ pip install github-reviewboard-sync
```

Usage

To use `github-reviewboard-sync` in a project:

```
import github-reviewboard-sync
```

Contributing

Contributions are welcome, and they are greatly appreciated! Every little bit helps, and credit will always be given. You can contribute in many ways:

3.1 Types of Contributions

3.1.1 Report Bugs

Report bugs at <https://github.com/timmartin19/github-reviewboard-sync/issues>.

If you are reporting a bug, please include:

- Your operating system name and version.
- Any details about your local setup that might be helpful in troubleshooting.
- Detailed steps to reproduce the bug.

3.1.2 Fix Bugs

Look through the GitHub issues for bugs. Anything tagged with “bug” is open to whoever wants to implement it.

3.1.3 Implement Features

Look through the GitHub issues for features. Anything tagged with “feature” is open to whoever wants to implement it.

3.1.4 Write Documentation

github-reviewboard-sync could always use more documentation, whether as part of the official github-reviewboard-sync docs, in docstrings, or even on the web in blog posts, articles, and such.

3.1.5 Submit Feedback

The best way to send feedback is to file an issue at <https://github.com/timmartin19/github-reviewboard-sync/issues>.

If you are proposing a feature:

- Explain in detail how it would work.
- Keep the scope as narrow as possible, to make it easier to implement.
- Remember that this is a volunteer-driven project, and that contributions are welcome :)

3.2 Get Started!

Ready to contribute? Here's how to set up *github-reviewboard-sync* for local development.

1. Fork the *github-reviewboard-sync* repo on GitHub.
2. Clone your fork locally:

```
$ git clone git@github.com:your_name_here/github-reviewboard-sync.git
```

3. Install your local copy into a virtualenv. Assuming you have virtualenvwrapper installed, this is how you set up your fork for local development:

```
$ mkvirtualenv github-reviewboard-sync
$ cd github-reviewboard-sync/
$ python setup.py develop
```

4. Create a branch for local development:

```
$ git checkout -b name-of-your-bugfix-or-feature
```

Now you can make your changes locally.

5. When you're done making changes, check that your changes pass flake8 and the tests, including testing other Python versions with tox:

```
$ flake8 github-reviewboard-sync tests
$ python setup.py test
$ tox
```

To get flake8 and tox, just pip install them into your virtualenv.

6. Commit your changes and push your branch to GitHub:

```
$ git add .
$ git commit -m "Your detailed description of your changes."
$ git push origin name-of-your-bugfix-or-feature
```

7. Submit a pull request through the GitHub website.

3.3 Pull Request Guidelines

Before you submit a pull request, check that it meets these guidelines:

1. The pull request should include tests.
2. If the pull request adds functionality, the docs should be updated. Put your new functionality into a function with a docstring, and add the feature to the list in README.rst.
3. The pull request should work for Python 2.6, 2.7, 3.3, 3.4 and 3.5, and for PyPy. Check https://travis-ci.org/timmartin19/github-reviewboard-sync/pull_requests and make sure that the tests pass for all supported Python versions.

3.4 Tips

To run a subset of tests:

```
$ python -m unittest tests.test_github-reviewboard-sync
```

Credits

4.1 Development Lead

- Apptimize - Tim Martin <martin@apptimize.com>

4.2 Contributors

None yet. Why not be the first?

History

5.1 0.1.6 (unreleased)

- Nothing changed yet.

5.2 0.1.5 (2016-06-03)

- Nothing changed yet.

5.3 0.1.4 (2016-05-09)

- Nothing changed yet.

5.4 0.1.3 (2016-05-06)

- Nothing changed yet.

5.5 0.1.2 (2016-05-05)

- First release on PyPI.

github-reviewboard-sync

Syncs pull requests with ReviewBoard submissions allowing you to create a pull request and review board submission at the same time

- Free software: MIT license
- Documentation: <https://github-reviewboard-sync.readthedocs.org>

6.1 Overview

This tool allows you to easily create a pull request and reviewboard submission at the same time

```
grs open my-feature-branch --github-username=MyUsername
```

You can also simply provide an environment variable `GITHUB_USERNAME` instead of passing in your github username. To add it to your bash profile simply run the following

```
echo 'export GITHUB_USERNAME=MyUsername' >> ~/.bash_profile
```

If you want to compare against a different branch and open a pull request to that branch

```
grs open my-feature-branch --base=version-branch
```

If you want to update an existing review board submission

```
grs open my-feature-branch -u
```

6.2 Installation

```
pip install github-reviewboard-sync
```

Or if you prefer

```
easy_install github-reviewboard-sync
```

6.3 Features

- Opens a pull request on github with a sane name and message based on commits

- Opens/updates a submission on review board with a sane summary and description based on commits
- Adds github pull request url to review board submission and vice versa

6.4 Credits

This package was created with [Cookiecutter](#) and the [audreyr/cookiecutter-pypackage](#) project template.

Indices and tables

- `genindex`
- `modindex`
- `search`