
Flask-Clacks Documentation

Release 1.0.0

William Mayor

Apr 26, 2017

Contents

1	Flask-Clacks	3
1.1	Features	3
1.2	Installation	3
1.3	Usage	3
1.4	Credits	4
2	Installation	5
2.1	Stable release	5
2.2	From sources	5
3	Contributing	7
3.1	Types of Contributions	7
3.2	Get Started!	8
3.3	Pull Request Guidelines	9
4	Credits	11
4.1	Development Lead	11
4.2	Contributors	11
5	History	13
5.1	1.0.0 (2017-04-26)	13
6	Indices and tables	15

Contents:

CHAPTER 1

Flask-Clacks

A man is not dead while his name is still spoken.

This is a very simple Flask extension that adds ‘X-Clacks-Overhead’ headers to your website’s responses.

- Free software: MIT license
- Documentation: <https://flask-clacks.readthedocs.io>.

Features

This package exposes a Flask extension which by default adds the header `X-Clacks-Overhead: GNU Terry Pratchett` on all routes, for all origins and methods.

- You can add extra names to your overhead
- You can decorate individual routes to have the overhead

Installation

Install the extension with using `pip`, or `easy_install`.

```
$ pip install -U flask-clacks
```

Usage

Apply to all routes, sending only Terry Pratchett’s name in the overhead.

```
from flask import Flask
from flask_clacks import Clacks

app = Flask(__name__)
Clacks(app)

@app.route("/")
def index():
    # Will have the header added to the response
    return "OK"
```

Apply to all routes, sending Terry Pratchett and John Dearheart's names in the overhead.

```
from flask import Flask
from flask_clacks import Clacks

app = Flask(__name__)
Clacks(app, names=('John Dearheart', ))

@app.route("/")
def index():
    # Will have the the clacks overhead header for both Terry and John
    return "OK"
```

Apply to specific routes, sending different names back on different responses.

```
from flask import Flask
from flask_clacks import clacks

app = Flask(__name__)

@app.route("/terry/")
@clacks
def terry():
    # Will have a clacks overhead header for Terry
    return "OK"

@app.route("/terry-and-john/")
@clacks(names=('John Dearheart', ))
def terry_and_john():
    # Will have a clacks overhead header for both Terry and John
    return "OK"

@app.route("/no-one/")
def no_one():
    # Will not have clacks overhead headers
    return "OK"
```

Credits

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

Thanks go to the [Flask-CORS](#) extension for providing decent examples of how to package an extension.

Stable release

To install Flask-Clacks, run this command in your terminal:

```
$ pip install -U flask-clacks
```

This is the preferred method to install Flask-Clacks, as it will always install the most recent stable release.

If you don't have [pip](#) installed, this [Python installation guide](#) can guide you through the process.

From sources

The sources for Flask-Clacks can be downloaded from the [Github repo](#).

You can either clone the public repository:

```
$ git clone git://github.com/WilliamMayor/flask_clacks
```

Or download the [tarball](#):

```
$ curl -OL https://github.com/WilliamMayor/flask_clacks/tarball/master
```

Once you have a copy of the source, you can install it with:

```
$ python setup.py install
```


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

Types of Contributions

Report Bugs

Report bugs at https://github.com/WilliamMayor/flask_clacks/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.

Fix Bugs

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

Implement Features

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

Write Documentation

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

Submit Feedback

The best way to send feedback is to file an issue at https://github.com/WilliamMayor/flask_clacks/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 :)

Get Started!

Ready to contribute? Here's how to set up *flask_clacks* for local development.

1. Fork the *flask_clacks* repo on GitHub.
2. Clone your fork locally:

```
$ git clone git@github.com:your_name_here/flask_clacks.git
```

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

```
$ cd flask_clacks/  
$ virtualenv venv  
$ . venv/bin/activate  
$ pip install . .[dev] .[test]
```

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:

```
$ make lint  
$ make test  
$ make test-all
```

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.

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 2.7 and 3.3+. Check https://travis-ci.org/WilliamMayor/flask_clacks/pull_requests and make sure that the tests pass for all supported Python versions.

CHAPTER 4

Credits

Development Lead

- William Mayor <mail@williammayor.co.uk>

Contributors

None yet. Why not be the first?

1.0.0 (2017-04-26)

- First release on PyPI.

CHAPTER 6

Indices and tables

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