

---

# **django-template***treeDocumentation*

***Release 0.1.1***

**Paul Butcher**

**Feb 02, 2018**



---

## Contents

---

<b>1</b>	<b>django-template_tree</b>	<b>3</b>
1.1	Documentation . . . . .	3
1.2	Quickstart . . . . .	3
1.3	Features . . . . .	4
1.4	Running Tests . . . . .	5
1.5	Credits . . . . .	5
<b>2</b>	<b>Installation</b>	<b>7</b>
<b>3</b>	<b>Usage</b>	<b>9</b>
<b>4</b>	<b>Contributing</b>	<b>13</b>
4.1	Types of Contributions . . . . .	13
4.2	Get Started! . . . . .	14
4.3	Pull Request Guidelines . . . . .	15
4.4	Tips . . . . .	15
<b>5</b>	<b>Credits</b>	<b>17</b>
5.1	Development Lead . . . . .	17
5.2	Contributors . . . . .	17
<b>6</b>	<b>History</b>	<b>19</b>
6.1	0.1.0 (2018-01-06) . . . . .	19



Contents:



Generate diagrams showing relationships between templates in Django projects

## 1.1 Documentation

The full documentation is at [https://django\\_template\\_tree.readthedocs.io](https://django_template_tree.readthedocs.io).

## 1.2 Quickstart

Install django-template\_tree:

```
pip install django_template_tree
```

Add it to your *INSTALLED\_APPS*:

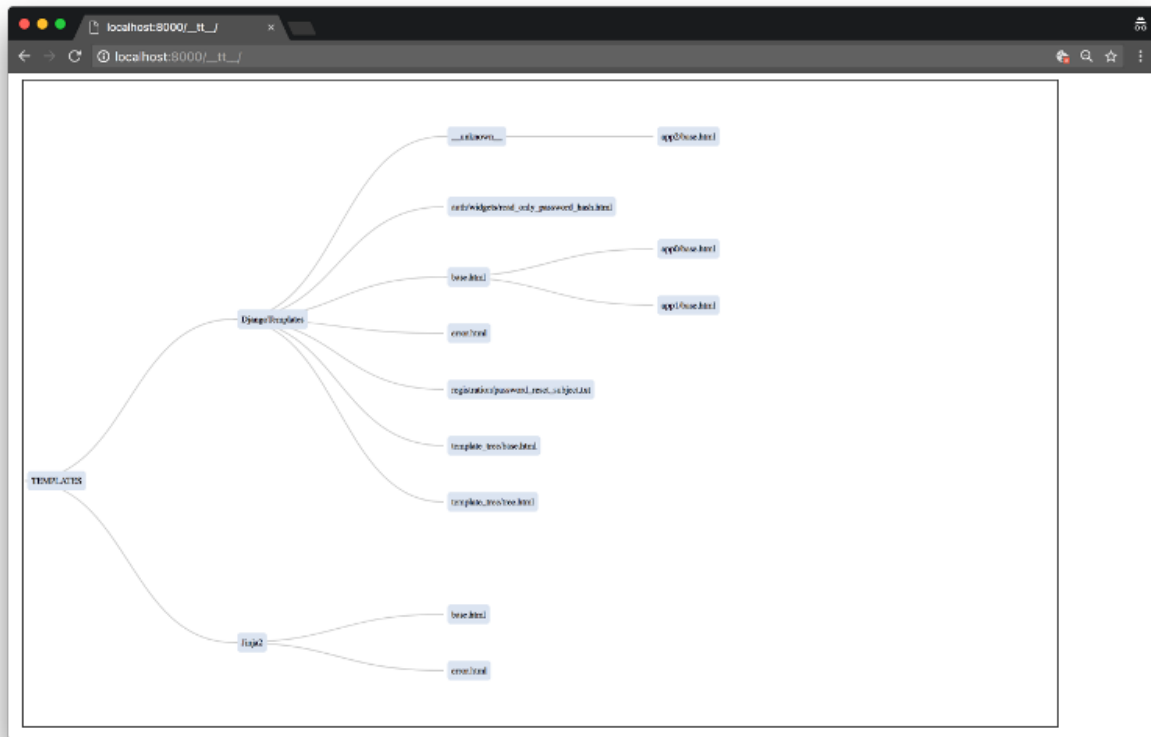
```
INSTALLED_APPS = (  
    ...  
    'template_tree',  
    ...  
)
```

Add django-template\_tree's URL patterns:

```
from template_tree import urls as template_tree_urls  
from django import settings  
if settings.DEBUG:  
    urlpatterns += [  
        url(r'^__tt__/', include('template_tree.urls', namespace='template_tree')),  
    ]
```

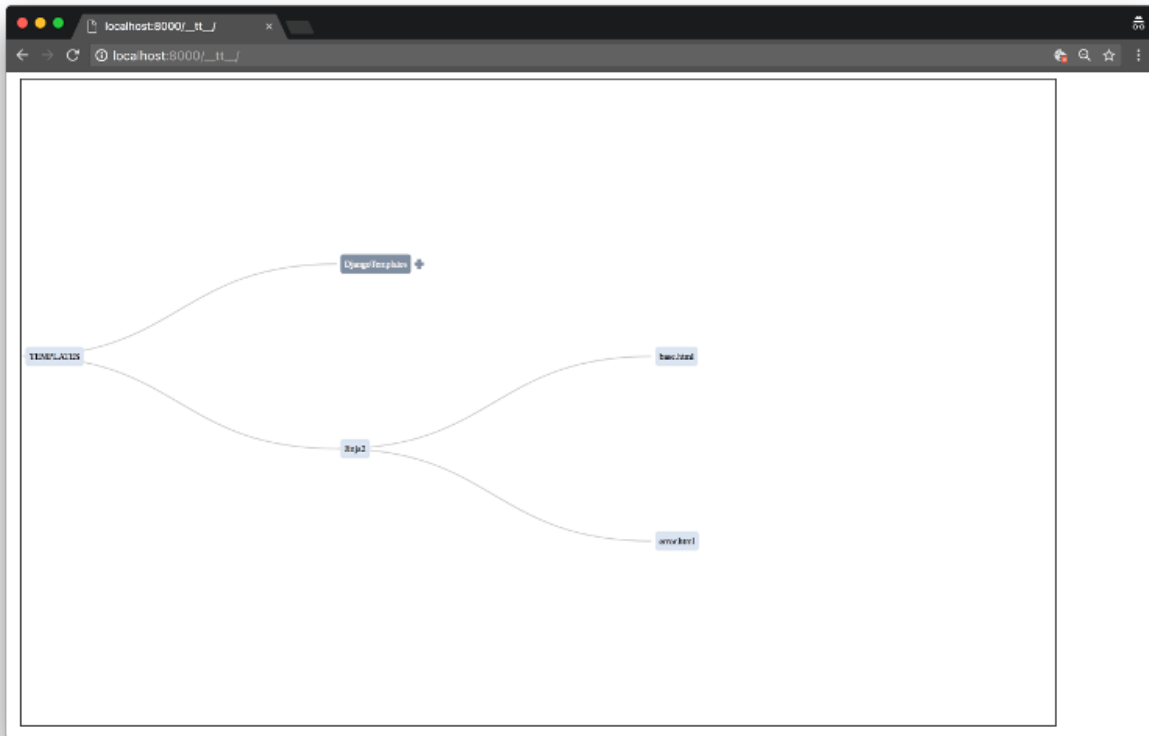
## 1.3 Features

Displays a collapsible tree diagram showing the hierarchy of templates used by a Django application.



Nodes can be collapsed and expanded, by clicking on them.





Apps can be filtered out of the tree, using the ‘exclude\_app’ querystring parameter, thus:

[http://localhost:8000/\\_\\_\\_tt\\_\\_\\_/?exclude\\_app=template\\_tree&exclude\\_app=admin](http://localhost:8000/___tt___/?exclude_app=template_tree&exclude_app=admin)

By default, admin is excluded, so [http://localhost:8000/\\_\\_\\_tt\\_\\_\\_/](http://localhost:8000/___tt___/) is equivalent to [http://localhost:8000/\\_\\_\\_tt\\_\\_\\_/?exclude\\_app=admin](http://localhost:8000/___tt___/?exclude_app=admin)

In order to show the admin app as well, you will need to ‘unexclude’ it thus: [http://localhost:8000/\\_\\_\\_tt\\_\\_\\_/?exclude\\_app=](http://localhost:8000/___tt___/?exclude_app=)

Visiting the template\_tree url, requesting an application/json response will return json data compatible with d3 hierarchies, so you can roll your own diagrams.

## 1.4 Running Tests

Does the code actually work?

```
source <YOURVIRTUALENV>/bin/activate
(myenv) $ pip install tox
(myenv) $ tox
```

## 1.5 Credits

Tools used in rendering this package:

- [Cookiecutter](#)

- `cookiecutter-djangopackage`

## CHAPTER 2

---

### Installation

---

At the command line:

```
$ easy_install django_template_tree
```

Or, if you have virtualenvwrapper installed:

```
$ mkvirtualenv django_template_tree  
$ pip install django_template_tree
```



## CHAPTER 3

---

### Usage

---

To use `django-template_tree` in a project, add it to your *INSTALLED\_APPS*:

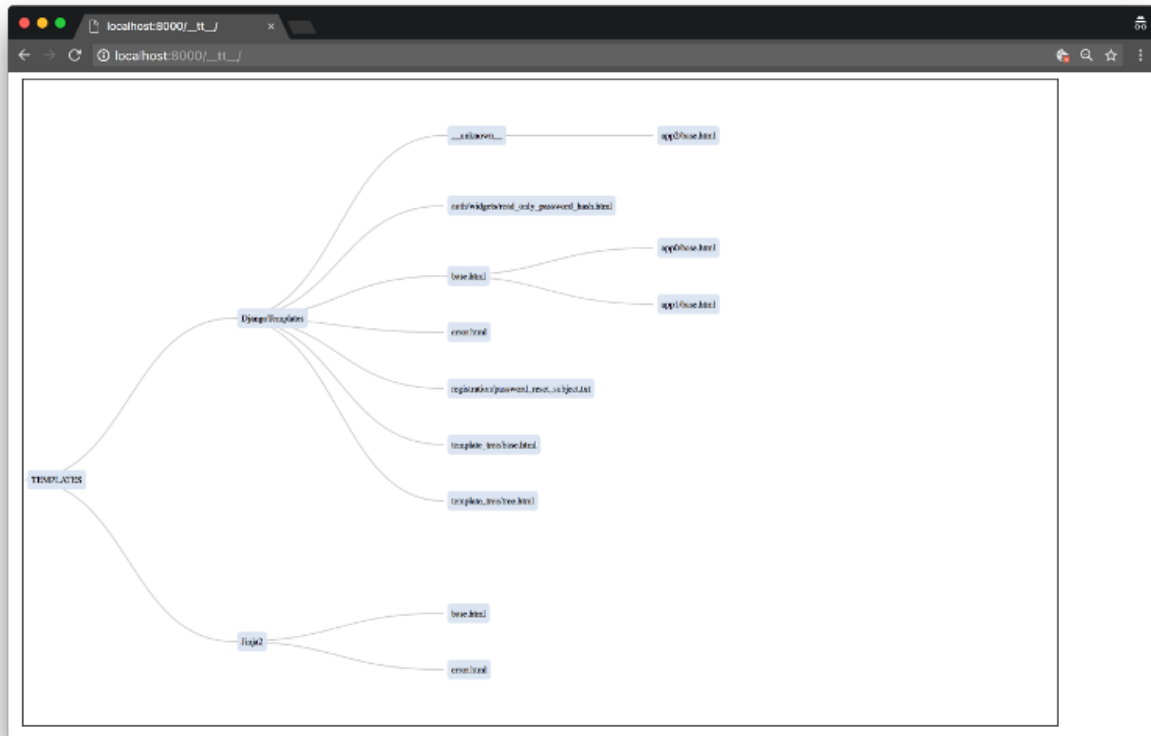
```
INSTALLED_APPS = (  
    ...  
    'template_tree',  
    ...  
)
```

Add `django-template_tree`'s URL patterns:

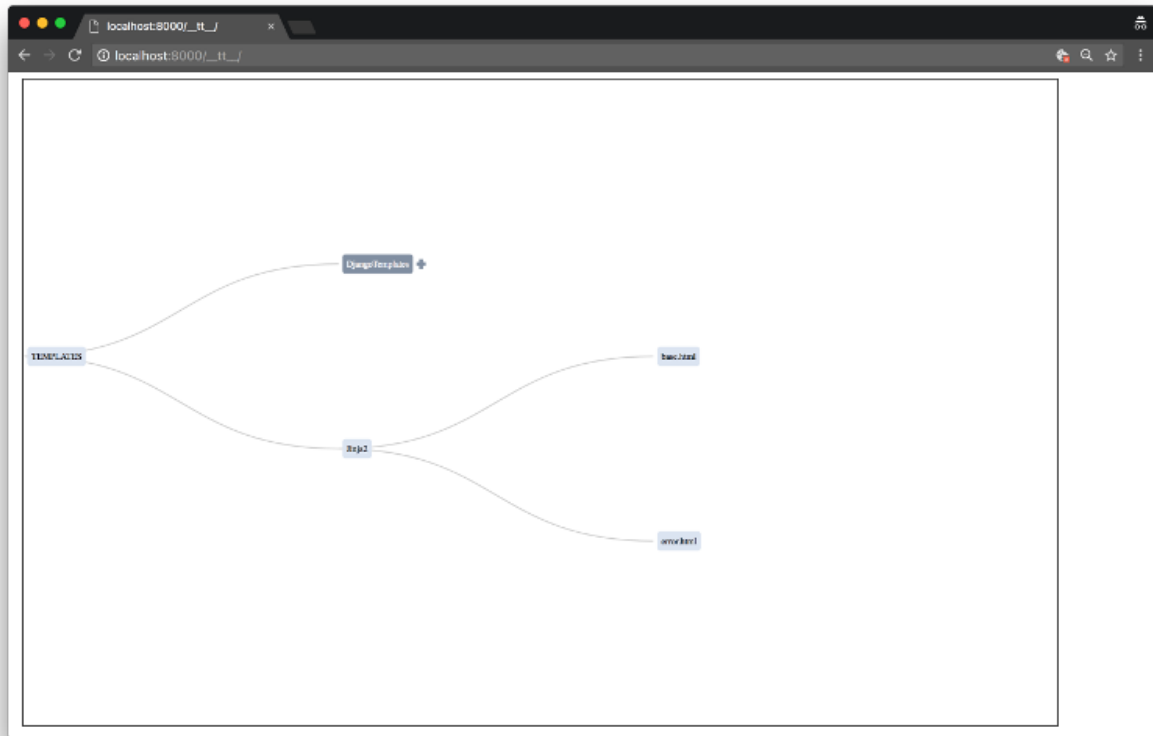
```
from template_tree import urls as django_template_tree_urls  
  
urlpatterns = [  
    ...  
    url(r'^__tt__/', include(django_template_tree_urls)),  
    ...  
]
```

Ideally, you should not enable this app in production, as it may reveal facts about your application that you wish to keep secret.

Visiting the URL in a browser will show a template inheritance diagram like this one:



Nodes can be collapsed (by clicking), in order to improve clarity or illustrate a particular line of inheritance.







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

## 4.1 Types of Contributions

### 4.1.1 Report Bugs

Report bugs at [https://github.com/paul-butcher/django\\_template\\_tree/issues](https://github.com/paul-butcher/django_template_tree/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.

### 4.1.2 Fix Bugs

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

### 4.1.3 Implement Features

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

### 4.1.4 Write Documentation

django-template<sub>tree</sub> could always use more documentation, whether as part of the official django-template<sub>tree</sub> docs, in docstrings, or even on the web in blog posts, articles, and such.

### 4.1.5 Submit Feedback

The best way to send feedback is to file an issue at [https://github.com/paul-butcher/django\\_template\\_tree/issues](https://github.com/paul-butcher/django_template_tree/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 :)

## 4.2 Get Started!

Ready to contribute? Here's how to set up *django\_template\_tree* for local development.

1. Fork the *django\_template\_tree* repo on GitHub.
2. Clone your fork locally:

```
$ git clone git@github.com:your_name_here/django_template_tree.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 django_template_tree
$ cd django_template_tree/
$ 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 django_template_tree 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.

## 4.3 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, and 3.3, and for PyPy. Check [https://travis-ci.org/paul-butcher/django\\_template\\_tree/pull\\_requests](https://travis-ci.org/paul-butcher/django_template_tree/pull_requests) and make sure that the tests pass for all supported Python versions.

## 4.4 Tips

To run a subset of tests:

```
$ python -m unittest tests.test_django_template_tree
```



### 5.1 Development Lead

- Paul Butcher <paul.butcher@gmail.com>

### 5.2 Contributors

None yet. Why not be the first?



#### 6.1 0.1.0 (2018-01-06)

- First release on PyPI.