django-crumbs Documentation

Release 1.0.0

Caktus Consulting Group

June 16, 2016

Contents

django-crumbs is a pluggable Django app for adding breadcrumbs to your project.

CHAPTER 1

Installation

1. Install django-crumbs with pip:

```
pip install django-crumbs
```

2. Add to your INSTALLED_APPS and run syncdb:

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

3. Make sure you have the "request" context processor in your config:

```
TEMPLATE_CONTEXT_PROCESSORS += ['django.core.context_processors.request']
```

CHAPTER 2

How it Works

django-crumbs provides two template tags. One (`add_crumb`) adds a breadcrumb to the current breadcrubs, the other (`render_breadcrumbs`) actually renders the accumulated breadcrumbs.

In your base template, you will generally include a template block that uses `add_crumb` to set up an initial first breadcrumb. You can then extend the breadcrumbs by defining the same block in child templates, using `{{ block.super }}` to maintain the content from parent templates, and adding additional breadcrumbs with additional `add_crumb` tags.

To render the accumulated breadcrumbs, include the `render_breadrubms` tag in the base template after the block which accumulates the breadcrumbs.

CHAPTER 3

Example

1. base.html:

```
<div id="breadcrumbs">
    {% block breadcrumb %}
        {% load breadcrumb_tags %}
        {% add_crumb 'Home' 'home' %}
    {% endblock %}
    {% render_breadcrumbs %}
</div>
```

2. person/list.html:

```
{% extends "base.html" %}
{% block breadcrumb %}
    {{ block.super }}
    {% load breadcrumb_tags %}
    {% add_crumb 'People' 'list_people' %}
{% endblock %}
```

3. person/view.html:

```
{% extends "person/list.html" %}
{% block breadcrumb %}
    {{ block.super }}
    {% load breadcrumb_tags %}
    {% add_crumb person.name 'view_person' person.pk %}
{% endblock %}
```

CHAPTER 4

Requirements

- django >= 1.3

Django 1.3 is the minimum level that is currently tested, it's likely that the django-crumbs code still works fine on earlier Django versions.

CHAPTER 5

Documentation

Full documentation for django-crumbs is availbe on Read the Docs.

CHAPTER 6

License

Development sponsored by Caktus Consulting Group, LLC.

CHAPTER 7

Documentation Contents

7.1 Template Tag Reference

django-crumbs provides two template tags: `add_crumb` and `render_breadcrumbs`. These template tags can be accessed by loading `breadcrumb_tags` in your template.

```
{% load breadcrumb_tags %}
```

7.1.1 add_crumb

The `add_crumb` tag may be used to add a breadcrumb either with or without an html anchor link. For a crumb without a link, simply provide the breadcrumb text.

```
{% add_crumb 'Home' %}
```

For a crumb with a link, provide the crumb text and either a url value or (better) the name of a url pattern to reverse in order to geneate the url.

```
{% add_crumb 'Home' '/home' %}
{% add_crumb 'Home' 'project-home' %}
```

The django-crumbs code assumes `url` arguments that contain slashes are already reversed and thus does not call reverse on these. However, it is generally better practice to avoid hard-coding url values in in templates, so this form should not be used unless absolutely necessary (if for some reason the target url value cannot be generated by a call to `reverse`).

The crumb and url arguments to the `add_crumb` tag may also be specified using keyword syntax.

```
{% add_crumb crumb='Home' url='project-home' %}
```

When passing a url pattern name to be reversed, additional arguments specified to `add_crumb` will be passed to the `reverse` call as `args`.

```
{% add_crumb 'Details' 'detail-view' 44 %}
```

The above call will result in a call to the `django.core.urlresolvers` `reverse` function:

```
reverse('detail-view', args=(44,))
```

The value returned from `reverse` will be the `href` value for the anchor link rendered with the breadcrumb.

Note: Additional arguments for the reverse call cannot be passed using keyword syntax.

7.1.2 render_breadcrumbs

The `render_breadcrumbs` tag renders the breadcrubs accumulated by previous calls to `add_crumb`, if there have been at least two calls to `add_crumb`. `render_breadcrumbs` takse no arguments.

The template used for rendering is `breadcrumbs/crubms.html`. django-crumbs provides a default template for this rendering in `crumbs/templates/breadcrumbs/crumbs.html`:

```
{% for crumb, href in crumbs %}
    {% if not forloop.last %}
        {% if href %}
            <a href='{{ href }}' title='Go back to {{ crumb }}'>{{ crumb }}</a> &raquo;
        {% else %}
            {{ crumb }} &raquo;
        {% endif %}
    {% else %}
        <span class='leaf'>{{ crumb }}</span>
    {% endif %}
{% endfor %}
```

If you would like to customize the rendering, you may override this template in your own project. The template is passed a sequence of `(crumb, href)` tuples in the `crumbs` context variable (crumbs without links will have empty `href` values in this sequence).

7.2 Contributing Guide

django-crumbs is a small, focused application that has been stable for a number of years. We do not anticipate any major feature additions or significant changes in future. However, if you encounter a bug or have an enhancement you would like to propose, then this guide will help you to contribute.

7.2.1 Ways to Contribute

You can contribute to the project by submitting bug reports, feature requests or documentation updates through the Github issues.

7.2.2 Getting the Source

You can clone the repository from Github:

```
git clone git://github.com/caktus/django-crumbs.git
```

However this checkout will be read only. If you want to contribute code you should create a fork and clone your fork. You can then add the main repository as a remote:

```
git clone git@github.com:<your-username>/django-crumbs.git
git remote add upstream git://github.com/caktus/django-crumbs.git
git fetch upstream
```

7.2.3 Running the Tests

When making changes to the code, either fixing bugs or adding features, you'll want to run the tests to ensure that you have not broken any of the existing functionality. With the code checked out and Django installed you can run the tests via:

```
python setup.py test
```

or:

```
python runtests.py
```

To test against multiple versions of Django you can use install and use `tox>=1.4`. The `tox` command will run the tests against Django 1.3, 1.4 and the current Git master using Python 2.6, plus currnet Git master with Python 3.2.:

```
# Build all environments
tox
# Build a single environment
tox -e py26-1.3.X
```

Building all environments will also build the documentation. More on that in the next section.

7.2.4 Building the Documentation

The docs are written in ReST and built using Sphinx. As noted above you can use tox to build the documentation or you can build them on their own via:

```
tox -e docs
```

or:

```
make html
```

from inside the `docs/` directory.

7.2.5 Coding Standards

Code contributions should follow the PEP8 and Django contributing style standards. Please note that these are only guidelines. Overall code consistency and readability are more important than strict adherence to these guides.

7.2.6 Submitting a Pull Request

The easiest way to contribute code or documentation changes is through a pull request. For information on submitting a pull request you can read the Github help page https://help.github.com/articles/using-pull-requests.

Pull requests are a place for the code to be reviewed before it is merged. This review will go over the coding style as well as if it solves the problem indended and fits in the scope of the project. It may be a long discussion or it might just be a simple thank you.

Not necessarily every request will be merged but you should not take it personally if you change is not accepted. If you want to increase the chances of your change being incorporated then here are some tips.

- Address a known issue. Preference is given to a request that fixes a currently open issue.
- Include documentation and tests when appropriate. New features should be tested and documented. Bugfixes should include tests which demostrate the problem.
- Keep it simple. It's difficult to review a large block of code so try to keep the scope of the change small.

You should also feel free to ask for help writing tests or writing documentation if you aren't sure how to go about it.

7.3 Release History

Release and change history for django-crumbs.

7.3.1 v1.0 (Not yet released)

Features

- Sphinx docs
- Standardized test setup, improved test coverage

7.3.2 v0.5 (Released 2010-12-04)

- Initial public release