Flask-Breadcrumbs Documentation

Release 0.4.0

Invenio Software

July 01, 2016
## Contents

1 Contents ............................................. 3  
1.1 Installation ......................................... 3  
1.2 Usage .............................................. 4  
1.3 Advanced Examples .................................. 6  
1.4 API ................................................... 7  
1.5 Changelog ........................................... 7  
1.6 Contributing ......................................... 7  
1.7 License ............................................... 8  
1.8 Authors ............................................. 8  

**Python Module Index** .................................... 11
Flask-Breadcrumbs is a Flask extension that adds support for generating site breadcrumb navigation.
1.1 Installation

Flask-Breadcrumbs is on PyPI so all you need is:

```bash
$ pip install Flask-Breadcrumbs
```

The development version can be downloaded from its page at GitHub.

```bash
$ git clone https://github.com/inveniosoftware/flask-breadcrumbs.git
$ cd flask-breadcrumbs
$ python setup.py develop
$ ./run-tests.sh
```

1.1.1 Requirements

Flask-Breadcrumbs has the following dependencies:

- Flask-Menu
- Flask
- six

Flask-Breadcrumbs requires Python version 2.6, 2.7 or 3.3+. 
1.2 Usage

This guide assumes that you have successfully installed Flask-Breadcrumbs package already. If not, please follow the Installation instructions first.

1.2.1 Simple Example

Here is a simple Flask-Breadcrumbs usage example:

```python
from flask import Flask
from flask_breadcrumbs import Breadcrumbs, register_breadcrumb

app = Flask(__name__)

# Initialize Flask-Breadcrumbs
Breadcrumbs(app=app)

@app.route('/', register_breadcrumb(app, '.', 'Home'))
def index():
    pass

if __name__ == '__main__':
    app.run(debug=True)
```

Save this as app.py and run it using your Python interpreter.

```
$ python app.py
* Running on http://127.0.0.1:5000/
```

1.2.2 Templating

By default, a proxy object to `current_breadcrumbs` is added to your Jinja2 context as `breadcrumbs` to help you with creating navigation bar. For example:

```html
<div>
  {$- for breadcrumb in breadcrumbs -$}
  <a href="{{ breadcrumb.url }}">{{ breadcrumb.text }}</a>
    {{ '/' if not loop.last }}
  {$- endfor -$}
</div>
```

1.2.3 Variable rules

For routes with a variable part, a dynamic list constructor can be used to create a more meaningful breadcrumb. In the example below, the User’s primary key is used to create a breadcrumb displaying their name.

```python
from flask import request, render_template

def view_user_dlc(*args, **kwargs):
    user_id = request.view_args['user_id']
    user = User.query.get(user_id)
    return [{'text': user.name, 'url': user.url}]
```
Flask-Breadcrumbs Documentation, Release 0.4.0

```python
@app.route('/users/<int:user_id>')
@breadcrumbs.register_breadcrumb(app, '.user.id', '',
    dynamic_list_constructor=view_user_dlc)
def view_user(user_id):
    user = User.query.get(user_id)
    return render_template('user.html', user=user)
```

### 1.2.4 Blueprint Support

The most important part of a modular Flask application is Blueprint. You can create one for your application somewhere in your code and decorate your view function, like this:

```python
from flask import Blueprint
from flask_breadcrumbs import register_breadcrumb

account = Blueprint('account', __name__, url_prefix='/account')

@account.route('/
@register_breadcrumb(account, '.', 'Your account')
def index():
    pass
```

### 1.2.5 Combining Multiple Blueprints

Sometimes you want to combine multiple blueprints and organise the navigation to certain hierarchy. This can be achieved by using the function `default_breadcrumb_root()`.

```python
from flask import Blueprint
from flask_breadcrumbs import default_breadcrumb_root, register_breadcrumb

social = Blueprint('social', __name__, url_prefix='/social')
default_breadcrumb_root(social, '.account')

@social.route('/list')
@register_breadcrumb(social, '.list', 'Social networks')
def list():
    pass
```

As a result of this, your `current_breadcrumbs` object will contain list with 3 items during processing request for `/social/list`.

```python
from example import app
from flask_breadcrumbs import current_breadcrumbs
import account
import social
app.register_blueprint(account.bp_account)
app.register_blueprint(social.bp_social)
with app.test_client() as c:
    c.get('/social/list')
    assert map(lambda x: x.url,
               list(current_breadcrumbs)) == [
        '/', '/account/', '/social/list']
```

### 1.2. Usage
1.3 Advanced Examples

1.3.1 Use with MethodViews and Blueprints

No routes are used in this example. Take note of the odd syntax for explicitly calling the decorator.

```python
from flask import Flask, render_template, Blueprint
from flask_breadcrumbs import Breadcrumbs, register_breadcrumb
from flask.views import MethodView

app = Flask(__name__)
Breadcrumbs(app=app)
bp = Blueprint('bp', __name__,)

class LevelOneView(MethodView):
    def get(self):
        return render_template('template.html')

class LevelTwoView(MethodView):
    def get(self):
        return render_template('template.html')

# Define the view by calling the decorator on its own,
# followed by the view inside parenthesis
level_one_view = register_breadcrumb(bp, 'breadcrumbs.', 'Level One')(LevelOneView.as_view('first'))
bp.add_url_rule('/one', view_func=level_one_view)  # Add the rule to the blueprint

level_two_view = breadcrumbs.register_breadcrumb(bp, 'breadcrumbs.two', 'Level Two')(LevelOneView.as_view('second'))
bp.add_url_rule('/two', view_func=level_two_view)
app.register_blueprint(bp)
```

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
</head>
<body>
    <h1>Example</h1>
    <div>
        {% for breadcrumb in breadcrumbs %}
        <a href="{{ breadcrumb.url }}">{{ breadcrumb.text }}</a>
        {% if not loop.last %}/{% endif %}
        {% endfor %}
    </div>
</body>
</html>
```
1.4 API

If you are looking for information on a specific function, class or method, this part of the documentation is for you.

1.4.1 Flask extension

1.4.2 Decorators

1.4.3 Proxies

```python
from flask_breadcrumbs import current_breadcrumbs

# List of breadcrumbs for current request.

from flask_breadcrumbs import breadcrumbs_root_path

# Name of breadcrumbs root path.
```

1.5 Changelog

Here you can see the full list of changes between each Flask-Breadcrumbs release.

Version 0.4.0 (released 2016-07-01)

- Removes support for Python 2.6.
- Adds an advanced example using MethodViews and Blueprints. (#23)
- Amends deprecated import of Flask extensions via `flask.ext`. (#29)

Version 0.3.0 (released 2015-03-16)

- Improved factory pattern support. (#19)
- Added example of using a dynamic list constructor with variables. (#16 #17)
- Allows usage of ordered breadcrumbs as menu. (#15)

Version 0.2.0 (released 2014-11-05)

- The Flask-Breadcrumbs extension is now released under more permissive Revised BSD License. (#11)
- Documentation improvements. (#13)
- Extension initialization improvements. (#12)
- Support for Python 3.4. (#5)

Version 0.1.0 (released 2014-07-24)

- Initial public release

1.6 Contributing

Bug reports, feature requests, and other contributions are welcome. If you find a demonstrable problem that is caused by the code of this library, please:

1. Search for already reported problems.
2. Check if the issue has been fixed or is still reproducible on the latest `master` branch.
3. Create an issue with a test case.

If you create a feature branch, you can run the tests to ensure everything is operating correctly:

```
$ ./run-tests.sh
```

## 1.7 License

Flask-Breadcrumbs is free software; you can redistribute it and/or modify it under the terms of the Revised BSD License quoted below.

Copyright (C) 2014, 2016 CERN.

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

In applying this license, CERN does not waive the privileges and immunities granted to it by virtue of its status as an Intergovernmental Organization or submit itself to any jurisdiction.

## 1.8 Authors

Flask-Breadcrumbs is developed for use in Invenio digital library software.

Contact us at info@inveniosoftware.org

- Krzysztof Lis <krzysztof.lis@cern.ch>
- Jiri Kuncar <jiri.kuncar@cern.ch>
- Tibor Simko <tibor.simko@cern.ch>
- Pierre Lucas <pierre.lucas@altie.fr>
- Joshua Arnott <josh@snorfalorpagus.net>
- Florian Merges <fmerges@fstarter.org>
• Nicholas Rutherford <zgithub@isomorphism.ca>
f

flask_breadcrumbs,7
Index

B
breadcrumbs_root_path (in module flask_breadcrumbs), 7

C
current_breadcrumbs (in module flask_breadcrumbs), 7

F
flask_breadcrumbs (module), 7