Wagtail Social Feed Documentation

Release 0.4.0

Tim Leguijt

Aug 16, 2017

Contents

Contents:

Wagtail Social Feed

A Wagtail module that provides pages and content blocks to show social media feeds

- Documentation: https://wagtailsocialfeed.readthedocs.io.

Features

- A wagtail settings sections to configure social media sources
- Social feed moderate view
- Social feed content block
- Social feed Page type

Implementations

The following social media sources are supported:

- Twitter
- Facebook
- Instagram

Installation

Stable release

To install Wagtail Social Feed, run this command in your terminal:

```
$ pip install wagtailsocialfeed
```

This is the preferred method to install Wagtail Social Feed, 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 Wagtail Social Feed can be downloaded from the Github repo.

You can either clone the public repository:

```
$ git clone git://github.com/LUKKIEN/wagtailsocialfeed
```

Or download the tarball:

```
$ curl  -OL https://github.com/LUKKIEN/wagtailsocialfeed/tarball/master
```

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

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

Configure Django

Add `wagtailsocialfeed` and `wagtail.contrib.modeladmin` to your `INSTALLED_APPS` in settings:

```
INSTALLED_APPS += [
    'wagtailsocialfeed',
    'wagtail.contrib.modeladmin',
]
```

Usage

Setting up social feeds

From the Wagtail CMS settings menu you can access the 'Social feeds' section. Add social media sources by defining the social media platform, the user to track and define if it's has to be moderated or not

Moderated feeds

When a social feed is marked as 'moderated' by default the latest posts of that feed are not visible to the visitors of your website. All posts have to be explicitally allowed before it will show up in a feed.

All moderated social feeds will show up as a new item in your CMS admin menu. From there you will have an overview of all the latest posts, search through the posts and add/remove posts from the moderated list.

Social feed page

It's easy to add a page to your tree which shows the latest posts of a specific feed. When adding a page just choose 'Social Feed Page' and select your feedconfig.

Styling the page

You can override the default template by creating a `wagtailsocialfeed/social_feed_page.html` in your templates directory. All items are available in the `{{ feed }}` variable.

Social feed blocks

You can also render the latest feed posts by using a `SocialFeedBlock` in your `StreamField`. Make sure you define the `SocialFeedBlock` as allowed block-type in the `StreamField`:

Styling the blocks

You can override the default template by creating a `wagtailsocialfeed/social_feed_block.html` in your templates directory. All items are available in the `{{ feed }}` variable.

Configuration

The following configuration options are available:

WAGTAIL_SOCIALFEED_CONFIG

The configuration for the social media accounts.

```
WAGTAIL_SOCIALFEED_CONFIG = {
    'twitter': {
        'CONSUMER_KEY': 'SOME_KEY',
        'CONSUMER_SECRET': 'SOME_SECRET',
        'ACCESS_TOKEN_KEY': 'SOME_KEY',
        'ACCESS_TOKEN_SECRET': 'SOME_SECRET'
    },
    'facebook': {
        'CLIENT_ID': 'SOME_ID',
        'CLIENT_SECRET': 'SOME_SECRET',
    }
}
```

No credentials are needed for Instagram.

Defaults to `{}`

WAGTAIL_SOCIALFEED_CACHE_DURATION

The cache timeout (in seconds) for the social feed items

Defaults to `900`

WAGTAIL_SOCIALFEED_SEARCH_MAX_HISTORY

The amount of time the module is allowed to search through the history of the social feed. This is only used for the moderator view. In the moderator view it is possible to enter a search query. For the search to return a usable data-set, multiple result-pages are requested from the social feed source. But it does need to have a limit.

Defaults to `timedelta(weeks=26)`

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:

Types of Contributions

Report Bugs

Report bugs at https://github.com/LUKKIEN/wagtailsocialfeed/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

Wagtail Social Feed could always use more documentation, whether as part of the official Wagtail Social Feed 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/LUKKIEN/wagtailsocialfeed/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 *wagtailsocialfeed* for local development.

1. Fork the *wagtailsocialfeed* repo on GitHub.

2. Clone your fork locally:

   ```
   $ git clone git@github.com:your_name_here/wagtailsocialfeed.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 wagtailsocialfeed
   $ cd wagtailsocialfeed/
   $ 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 wagtailsocialfeed tests
   $ python setup.py test or 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.

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, 3.3, 3.4 and 3.5 and for PyPy. Check https://travis-ci.org/tleguijt/wagtailsocialfeed/pull_requests and make sure that the tests pass for all supported Python versions.

Tips

To run a subset of tests:

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

CHAPTER 6

Indices and tables

- genindex
- modindex