django-hitcount Documentation

Release 1.1.0

Damon Timm

July 04, 2015

Contents

Django-Hitcount allows you to track the number of hits/views for a particular object.

CHAPTER 1

Overview

Django-Hitcount allows you to track the number of hits (views) for a particular object. This isn't meant to be a full-fledged tracking application or a real analytic tool; it's just a basic hit counter.

How one tracks a "hit" or "view" of a web page is not such a simple thing as it might seem. That's why folks rely on Google Analytics or similiar tools. It's tough! This is a simple app with some settings and features that should suit the basic needs of smaller sites.

It comes ready to track hits (to use the out-of-the-box method, you will need jQuery – although writing your own JavaScript implementation is not hard).

1.1 Requirements and Compatibility

Currently supporting Django >= 1.4 and any Python version supported by your Django version.

CHAPTER 2

Installation and Usage

For a working implementation, you can view the example project on Github.

Install django-hitcount:

```
pip install django-hitcount
```

2.1 Settings.py

Add django-hitcount to your `INSTALLED_APPS`, enable `SESSION_SAVE_EVERY_REQUEST`:

```
# settings.py

INSTALLED_APPS = (
    ...
    'hitcount'
)

# needed for django-hitcount to function properly
SESSION_SAVE_EVERY_REQUEST = True
```

2.2 Urls.py

In your `urls.py` file add the following:

```
# urls.py
urlpatterns = patterns('',
    ...
    url(r'hitcount/', include('hitcount.urls', namespace='hitcount')),
)
```

View the additional settings section for more information.

2.3 Template Magic

Django-Hitcount can utilize Javascript out-of-the-box to record the `Hits` to an object (be it a blog post, poll, etc). There is a jQuery implementation included with the app though you could write your own implementation, or copy-paste the one included, easily enough.

Start by loading hitcount tags on the desired templates:

```
{% load hitcount_tags %}
```

2.3.1 Recording a Hit

If you want to use the jQuery implementation in your project, you can add the Javascript file to your template like so:

```
{% load staticfiles %}
<script src="{% static 'hitcount/hitcount-jquery.js' %}"></script>
```

Then, on your object detail page (blog, page, poll, etc) you inject the needed javascript variables:

```
# use default insertion method for hitcount-jquery.js:
{% insert_hit_count_js_variables for object %}

# or you can use a template variable to inject as you see fit
{% get_hit_count_js_variables for object as hitcount %}
({ hitcount.ajax_url }}
{{ hitcount.pk }}
```

2.3.2 Displaying Hit Information

You can retrieve the number of hits for an object many different ways:

```
# Return total hits for an object:
{% get_hit_count for [object] %}

# Get total hits for an object as a specified variable:
{% get_hit_count for [object] as [var] %}

# Get total hits for an object over a certain time period:
{% get_hit_count for [object] within ["days=1,minutes=30"] %}

# Get total hits for an object over a certain time period as a variable:
{% get_hit_count for [object] within ["days=1,minutes=30"] as [var] %}
```

CHAPTER 3

Additional Settings

There are a few additional settings you can use to customize django-hitcount and are set in your `settings.py` file.

3.1 HITCOUNT_KEEP_HIT_ACTIVE

This is the number of days, weeks, months, hours, etc (using a `timedelta` keyword argument), that an `Hit` is kept **active**. If a `Hit` is **active** a repeat viewing will not be counted. After the **active** period ends, however, a new `Hit` will be recorded. You can decide how long you want this period to last and it is probably a matter of preference.:

```
# default value
HITCOUNT_KEEP_HIT_ACTIVE = { 'days': 7 }
```

3.2 HITCOUNT_HITS_PER_IP_LIMIT

Limit the number of **active** `Hits` from a single IP address. 0 means that it is unlimited.:

```
# default value
HITCOUNT_HITS_PER_IP_LIMIT = 0
```

3.3 HITCOUNT_EXCLUDE_USER_GROUP

Exclude `Hits` from all users in the specified user groups. By default, this is set to an empty list (all users counted). In the example, below, it will exclude all your 'Editors'.:

```
# example value, default is empty tuple
HITCOUNT_EXCLUDE_USER_GROUP = ( 'Editor', )
```

3.4 HITCOUNT_KEEP_HIT_IN_DATABASE

This setting is used with the `hitcount_cleanup` management command and specifies a `timedelta` within which to keep/save `Hits`. Any `Hit` older than the time specified will be removed for the `Hits` table.:

```
# default value
HITCOUNT_KEEP_HIT_IN_DATABASE = { 'days': 30 }
```

CHAPTER 4

Management Commands

If you would like to periodically prune your stale `Hits` you can do so by running the the management command `hitcount_cleanup`.:

```
./manage.py hitcount_cleanup
```

The command relies on the setting `HITCOUNT_KEEP_HIT_IN_DATABASE` to determine how far back to prune. See the additional settings section for more information.

CHAPTER 5

Example Project

There is an example project the demonstrates counting hits on a blog post using JavaScript on GitHub. It's fairly easy to get this working using the Django development server. Be sure to run this inside your own `virtualenv` (but who doesn't, these days?!).:

```
$ git clone git@github.com:thornomad/django-hitcount.git
$ cd django-hitcount/example_project
$ pip install -r requirements.txt   # sqlite requires pytz
$ python manage.py migrate          # will load some data fixtures for you
$ python manage.py createsuperuser  # for access to the admin portion
$ python manage.py runserver        # should be all set!
```

You can open your favorite browser console and see some of the Javascript `console.log` output indicating what type of `Hit` is being counted (or ignored). When you are ready to work on your own site, check out the Installation and Usage and Additional Settings sections.

CHAPTER 6

Contribution and Testing

I would love to make it better. Please fork, branch, and push. I plan to do my work in the `develop` branch before moving it to `master` for a real release. You can safely ignore the `devel` branch which is old and stale but has something in it I can't remember why I'm saving.

6.1 Testing

You can run the tests by installing the requirements and then executing `runtests.py`:

```
$ pip install -r tests/requirements.txt
$ ./runtests.py
```

This method using `py.test` for test discovery (so older versions of Django can find all the tests). If you would like to use Django's own test runner you can execute:

```
$ ./runtests.py --django
```

CHAPTER 7

Additional Authors and Thanks

I've had some help and I'm very grateful! You can always look at the contributors on GitHub to get a clearer picture. This doesn't include everyone and if I missed someone let me know I will add it.

Thanks goes to:

- Basil Shubin and his work at django-hitcount-headless as well as his Russian translations
- ariddell for putting the *setup.py* package together for me

CHAPTER 8

Changelog

8.1 Version 1.1.0:

- added tests (lots of them)
- added documentation
- support for Django 1.4.x - 1.8.x
- support for Python 3.x
- created an example project
- squashed bugs
- released to pip
- more, I'm sure!

CHAPTER 9

Issues

Use the GitHub issue tracker for django-hitcount to submit bugs, issues, and feature requests.