
django-thummer Documentation

Release 2.0.6

Matt Austin

Feb 07, 2018

Contents

1	Setup	3
1.1	Requirements	3
1.2	Installation	3
2	Basic API Example	5
3	Template Tag	7
4	Settings	9
4.1	THUMMER_PLACEHOLDER_DEFAULTS	9
4.2	THUMMER_PLACEHOLDER_PATH	9
4.3	THUMMER_QUEUE_SNAPSHOTS	9
4.4	THUMMER_SNAPSHOTS_VALID_FOR	9
4.5	THUMMER_STORAGE	10
4.6	THUMMER_THUMBNAIL_DEFAULTS	10
4.7	THUMMER_UPLOAD_PATH	10
5	Credits	11
6	Indices and tables	13

A website screenshot and thumbnailing app for [Django](#). Uses firefox, selenium, sorl-thumbnail, and celery.

Contents:

1.1 Requirements

django-thummer requires Firefox and Xvfb. On Debian/Ubuntu systems these can be installed using:

```
sudo apt-get install firefox xvfb
```

1.2 Installation

django-thummer can be installed to your Python environment using pip:

```
pip install django-thummer
```

Then add `thummer` to `INSTALLED_APPS` in your Django settings file, and sync the database. If you're using [South](#), then migrations are provided.

1.2.1 Using Celery for background thumbnail capture

It is highly recommended to use Celery for background thumbnail capture. By default, django-thummer will automatically use Celery if it is available. Both `celery` and `django-celery` can be installed to your Python environment using pip:

```
pip install django-celery
```

Configuring celery can be tricky, and you should carefully read both the [django-celery documentation](#) and [celery documentation](#).

CHAPTER 2

Basic API Example

```
from thummer import get_thumbnail  
  
thumbnail = get_thumbnail('http://www.example.com/', '400x400')
```


CHAPTER 3

Template Tag

```
{% load thummer %}

{% thummer "http://www.example.com/" "400x400" as thumb %}

{% endthummer %}
```

Reference:

4.1 THUMMER_PLACEHOLDER_DEFAULTS

Default: `{'crop': 'center', 'upscale': False, 'format': 'PNG'}` (Dictionary)

A keywords dictionary passed to `sorl.thumbnail` when generating placeholder thumbnails.

4.2 THUMMER_PLACEHOLDER_PATH

Default: `'thummer/placeholder.png'` (String)

The path to the placeholder image in your `staticfiles` storage.

4.3 THUMMER_QUEUE_SNAPSHOTS

Default: `True` (Boolean) if `'djcelery'` is in Django's `'INSTALLED_APPS'` setting.

Determine if thumbnail generation should be queued for background processing using `django-celery` (recommended).

4.4 THUMMER_SNAPSHOTS_VALID_FOR

Default: `dateutil.relativedelta.relativedelta(months=+3)` (A `dateutil.relativedelta` of 3 months)

The period of time that a thumbnail snapshot of a url is valid for, before a new thumbnail is generated.

4.5 THUMMER_STORAGE

Default: `django.core.files.storage.default_storage` (Django's configured default storage)

Storage instance to use for thumbnail media.

4.6 THUMMER_THUMBNAIL_DEFAULTS

Default: `{'crop': 'left top', 'upscale': False, 'format': 'JPEG'}` (Dictionary)

A keywords dictionary passed to `sorl.thumbnail` when generating thumbnails.

4.7 THUMMER_UPLOAD_PATH

Default: `'thummer/snapshots'` (String)

The path prefix to store thumbnails under in `THUMMER_STORAGE`.

CHAPTER 5

Credits

Special thanks to the [sorl-thumbnail](#), [selenium](#), [PyVirtualDisplay](#), and [celery](#), projects. Thanks also to the blog posts which originally got me thinking about the project:

- <http://coreygoldberg.blogspot.com/2011/06/python-headless-selenium-webdriver.html>
- <http://lapin-blanc.net/09/11/2008/django-website-thumbnail-generator/>

CHAPTER 6

Indices and tables

- `genindex`
- `modindex`
- `search`