

---

# **django-dummy-thumbnails**

## **Documentation**

*Release 0.1.10*

**Artur Barseghyan <[artur.barseghyan@gmail.com](mailto:artur.barseghyan@gmail.com)>**

January 26, 2017



<b>1</b>	<b>Prerequisites</b>	<b>3</b>
<b>2</b>	<b>Installation</b>	<b>5</b>
<b>3</b>	<b>Usage</b>	<b>7</b>
3.1	Replace broken images with dummy ones . . . . .	7
3.1.1	Supported thumbnailers . . . . .	7
	django-imagekit . . . . .	7
	sorl-thumbnail . . . . .	8
	easy-thumbnails . . . . .	8
<b>4</b>	<b>Dealing with broken or invalid dummy images</b>	<b>11</b>
<b>5</b>	<b>Demo</b>	<b>13</b>
5.1	Run demo locally . . . . .	13
<b>6</b>	<b>Testing</b>	<b>15</b>
<b>7</b>	<b>License</b>	<b>17</b>
<b>8</b>	<b>Support</b>	<b>19</b>
<b>9</b>	<b>Author</b>	<b>21</b>
<b>10</b>	<b>Documentation</b>	<b>23</b>
10.1	dummy_thumbnails package . . . . .	23
10.1.1	Subpackages . . . . .	23
	dummy_thumbnails.contrib package . . . . .	23
	Subpackages . . . . .	23
	dummy_thumbnails.contrib.image_importers package . . . . .	23
	Subpackages . . . . .	23
	Module contents . . . . .	24
	dummy_thumbnails.contrib.thumbnailers package . . . . .	24
	Subpackages . . . . .	24
	dummy_thumbnails.contrib.thumbnailers.django_imagekit package . . . . .	24
	Submodules . . . . .	24
	dummy_thumbnails.contrib.thumbnailers.django_imagekit.generatorlibrary module . . . . .	24
	Module contents . . . . .	24
	dummy_thumbnails.contrib.thumbnailers.easy_thumbnails package . . . . .	24

Submodules . . . . .	24
dummy_thumbnails.contrib.thumbnailers.easy_thumbnails.source_generators module . . . . .	24
Module contents . . . . .	24
dummy_thumbnails.contrib.thumbnailers.sorl_thumbnail package . . . . .	24
Subpackages . . . . .	24
dummy_thumbnails.contrib.thumbnailers.sorl_thumbnail.engines package . . . . .	24
Submodules . . . . .	24
dummy_thumbnails.contrib.thumbnailers.sorl_thumbnail.engines.dummy_engine module . . . . .	24
Module contents . . . . .	24
Module contents . . . . .	24
Module contents . . . . .	24
Module contents . . . . .	25
dummy_thumbnails.management package . . . . .	25
Subpackages . . . . .	25
dummy_thumbnails.management.commands package . . . . .	25
Submodules . . . . .	25
dummy_thumbnails.management.commands.dummy_thumbnails_symlink_dummy_images module . . . . .	25
dummy_thumbnails.management.commands.dummy_thumbnails_verify_dummy_images module . . . . .	25
Module contents . . . . .	25
Module contents . . . . .	25
dummy_thumbnails.tests package . . . . .	25
Submodules . . . . .	25
dummy_thumbnails.tests.base module . . . . .	25
dummy_thumbnails.tests.helpers module . . . . .	26
dummy_thumbnails.tests.test_core module . . . . .	26
Module contents . . . . .	26
10.1.2 Submodules . . . . .	26
10.1.3 dummy_thumbnails.apps module . . . . .	26
10.1.4 dummy_thumbnails.base module . . . . .	26
10.1.5 dummy_thumbnails.conf module . . . . .	26
10.1.6 dummy_thumbnails.defaults module . . . . .	27
10.1.7 dummy_thumbnails.helpers module . . . . .	27
10.1.8 dummy_thumbnails.settings module . . . . .	27
10.1.9 Module contents . . . . .	27

<b>11 Indices and tables</b>	<b>29</b>
------------------------------	-----------

<b>Python Module Index</b>	<b>31</b>
----------------------------	-----------

Dummy thumbnails for *most popular* Django thumbnail generators.

There are times when you have a database of a Django site and you need to quickly start it up to fix/develop, but then you realise that images are missing and you need to have images, because either your layout is broken or, in the worst case, entire site is broken (500). This library has been written for developers in order to avoid above mentioned problems in the shortest way possible, with least efforts possible.



---

### Prerequisites

---

- Django 1.8, 1.9, 1.10
- Python 2.7, 3.4, 3.5

Although *django-dummy-thumbnails* is not being tested against older versions of Django, tests do pass with Django versions 1.5, 1.6 and 1.7.





---

## Installation

---

### 1. Install in your virtual environment

Latest stable version from PyPI:

```
pip install django-dummy-thumbnails
```

Latest stable version from GitHub:

```
pip install https://github.com/barseghyanartur/django-dummy-thumbnails/archive/stable.tar.gz
```

### 2. Add `dummy_thumbnails` to your `INSTALLED_APPS` in the `global settings.py`.

```
INSTALLED_APPS = (  
    # ...  
    'dummy_thumbnails',  
    # ...  
)
```

### 3. Specify a custom path to your images directory:

```
DUMMY_THUMBNAILS_IMAGES_PATH = '/home/path/to/images/'
```

This should be a directory inside your media directory of your Django project. Otherwise Django would raise a `SuspiciousOperation` exception. In order not to duplicate tons of files for each project, you are advised to create symlinks to the images directory in the media directory of your Django project.

```
ln -s /home/path/to/images /home/repos/your-django-project/media
```

If you prefer to use included **public domain images**, run the following management commands:

```
./manage.py collectstatic --noinput  
./manage.py dummy_thumbnails_symlink_dummy_images
```

And specify the following path:

```
DUMMY_THUMBNAILS_IMAGES_PATH = os.path.join(MEDIA_ROOT, 'mixed')
```



---

## Usage

---

Common usage examples.

### 3.1 Replace broken images with dummy ones

That's what it's all about - replacing the broken images with dummy ones.

---

**Note:** You should **never** use this in production. All the changes mentioned above and below are supposed to be applied to **development** settings only.

---

#### 3.1.1 Supported thumbnailers

Most popular image thumbnailers for Django ([django-imagekit](#), [sorl-thumbnail](#) and [easy-thumbnails](#)) are supported. If you can't find your favourite thumbnailer, open an issue or consider making a pull request.

##### django-imagekit

Integration with [django-imagekit](#).

Modify your settings in the following way:

1. Add `imagekit`, `dummy_thumbnails` and `dummy_thumbnails.contrib.thumbnailers.django_imagekit.generatorlibrary` to the `INSTALLED_APPS`:

```
INSTALLED_APPS = [  
    # ...  
    'imagekit',  
    'dummy_thumbnails',  
    'dummy_thumbnails.contrib.thumbnailers.django_imagekit.generatorlibrary',  
    # ...  
]
```

2. If you are using the included public domain images, don't forget to collect the static files and create a symlink:

```
./manage.py collectstatic --noinput  
./manage.py dummy_thumbnails_symlink_dummy_images
```

3. Now the following would work:

```
{% load imagekit %}

{% thumbnail '640x480' 'None1' %}
{% thumbnail '480x640' 'None2' %}
{% thumbnail '200x200' 'None3' %}
```

## sorl-thumbnail

Integration with [sorl-thumbnail](#).

Modify your settings in the following way:

1. Add `sorl.thumbnail` and `dummy_thumbnails` to the `INSTALLED_APPS`:

```
INSTALLED_APPS = [
    # ...
    'sorl.thumbnail',
    'dummy_thumbnails',
    # ...
]
```

2. Set the dummy thumbnail engine as `THUMBNAIL_ENGINE`:

```
THUMBNAIL_ENGINE = 'dummy_thumbnails.contrib.sorl_thumbnail.engines.DummyThumbnailsEngine'
```

3. If you are using the included public domain images, don't forget to collect the static files and create a symlink:

```
./manage.py collectstatic --noinput
./manage.py dummy_thumbnails_symlink_dummy_images
```

4. Now the following would work:

```
{% load thumbnail %}

{% thumbnail 'None1' "640x480" crop="center" as im %}

{% endthumbnail %}

{% thumbnail 'None2' "480x640" crop="center" as im %}

{% endthumbnail %}

{% thumbnail 'None3' "200x200" crop="center" as im %}

{% endthumbnail %}
```

## easy-thumbnails

Integration with [easy-thumbnails](#).

Modify your settings in the following way:

1. Add `easy_thumbnails` and `dummy_thumbnails` to the `INSTALLED_APPS`:

```
INSTALLED_APPS = [
    # ...
    'easy_thumbnails',
    'dummy_thumbnails',
]
```

```
    # ...  
]
```

2. Add dummy thumbnail generator to THUMBNAIL\_SOURCE\_GENERATORS:

```
THUMBNAIL_SOURCE_GENERATORS = (  
    'dummy_thumbnails.contrib.thumbnailers.easy_thumbnails.source_generators.dummy_thumbnail',  
)
```

3. If you are using the included public domain images, don't forget to collect the static files and create a symlink:

```
./manage.py collectstatic --noinput  
./manage.py dummy_thumbnails_symlink_dummy_images
```

4. Now the following would work:

```
{% load thumbnail %}  
  
  
  

```



---

## Dealing with broken or invalid dummy images

---

Of course, it's always better to have a good working set of dummy images. However, it might happen that for some reason one of your dummy images is broken.

The recommended approach is to use a management command `dummy_thumbnails_verify_dummy_images`, which has been written in order to verify the dummy images and identify possible problems. It also lets you remove broken/invalid dummy images.

To remove broken/invalid dummy images with confirmation, type:

```
./manage.py dummy_thumbnails_verify_dummy_images
```

To remove broken/invalid dummy images without confirmation, type:

```
./manage.py dummy_thumbnails_verify_dummy_images --noinput
```

To just list broken/invalid dummy images without removal, type:

```
./manage.py dummy_thumbnails_verify_dummy_images --list
```

Another way to avoid failures is to set the value of `DUMMY_THUMBNAILS_VERIFY_IMAGES` to `True` in your project settings. Beware, that this slows down the start up time of your Django project, although does not slow down further rendering of the images.





---

## Demo

---

### 5.1 Run demo locally

In order to be able to quickly evaluate the *django-dummy-thumbnails*, a demo app (with a quick installer) has been created (works on Ubuntu/Debian, may work on other Linux systems as well, although not guaranteed). Follow the instructions below to have the demo running within a minute.

Grab the latest `dummy_thumbnails_demo_installer.sh`:

```
wget -O - https://raw.githubusercontent.com/barseghyanartur/django-dummy-thumbnails/stable/examples/dummy_thumbnails_demo_installer.sh
```

Open your browser and test the app.

- URL: <http://127.0.0.1:8001/>

If quick installer doesn't work for you, see the manual steps on running the [example project](#).



---

## Testing

---

Simply type:

```
./runtests.py
```

or use tox:

```
tox
```

or use tox to check specific env:

```
tox -e py35
```

or run Django tests:

```
./manage.py test dummy_thumbnails --settings=settings.testing
```



---

**License**

---

GPL 2.0/LGPL 2.1



---

**Support**

---

For any issues contact me at the e-mail given in the *Author* section.





---

**Author**

---

Artur Barseghyan <[artur.barseghyan@gmail.com](mailto:artur.barseghyan@gmail.com)>



---

## Documentation

---

Contents:

### Table of Contents

- *django-dummy-thumbnails*
  - *Prerequisites*
  - *Installation*
  - *Usage*
    - \* *Replace broken images with dummy ones*
      - *Supported thumbnailers*
      - *django-imagekit*
      - *sorl-thumbnail*
      - *easy-thumbnails*
  - *Dealing with broken or invalid dummy images*
  - *Demo*
    - \* *Run demo locally*
  - *Testing*
  - *License*
  - *Support*
  - *Author*
  - *Documentation*
  - *Indices and tables*

## 10.1 dummy\_thumbnails package

### 10.1.1 Subpackages

**dummy\_thumbnails.contrib package**

**Subpackages**

**dummy\_thumbnails.contrib.image\_importers package**

**Subpackages**

**Module contents**

**dummy\_thumbnails.contrib.thumbnailers package**

**Subpackages**

**dummy\_thumbnails.contrib.thumbnailers.django\_imagekit package**

**Submodules**

**dummy\_thumbnails.contrib.thumbnailers.django\_imagekit.generatorlibrary module**

**Module contents**

**dummy\_thumbnails.contrib.thumbnailers.easy\_thumbnails package**

**Submodules**

**dummy\_thumbnails.contrib.thumbnailers.easy\_thumbnails.source\_generators module**

**Module contents**

**dummy\_thumbnails.contrib.thumbnailers.sorl\_thumbnail package**

**Subpackages**

**dummy\_thumbnails.contrib.thumbnailers.sorl\_thumbnail.engines package**

**Submodules**

**dummy\_thumbnails.contrib.thumbnailers.sorl\_thumbnail.engines.dummy\_engine module**

**Module contents**

**Module contents**

**Module contents**

## Module contents

### dummy\_thumbnails.management package

#### Subpackages

#### dummy\_thumbnails.management.commands package

#### Submodules

**dummy\_thumbnails.management.commands.dummy\_thumbnails\_symlink\_dummy\_images** module Management command for creating symlinks.

- Command: Management command. Prepare the dirs and create symlinks for testing purposes.

**class** `dummy_thumbnails.management.commands.dummy_thumbnails_symlink_dummy_images.Command` (*stdou*  
*stder*  
*no\_co*

Bases: `django.core.management.base.BaseCommand`

Prepare dirs and symlinks.

**handle** (*\*args, \*\*options*)  
Handle.

**help** = 'Prepare dirs and make symlinks to dummy images'

#### dummy\_thumbnails.management.commands.dummy\_thumbnails\_verify\_dummy\_images module

## Module contents

## Module contents

### dummy\_thumbnails.tests package

#### Submodules

#### dummy\_thumbnails.tests.base module

Base tests module.

- `log_info`: Method decorator, logs info about the method (name, return value).
- `LOG_INFO`: Flag.
- `app_setup`: Setup the app.
- `skip`: Method decorator. If used, tests is skipped.
- `is_app_setup_completed`: Check if app setup has been completed.
- `mark_app_setup_as_completed`: Mark the app setup as completed.

`dummy_thumbnails.tests.base.log_info` (*func*)  
Log some useful info.

`dummy_thumbnails.tests.base.skip(func)`

Simply skip the test.

`dummy_thumbnails.tests.base.is_app_setup_completed()`

Check if app setup is completed.

`dummy_thumbnails.tests.base.mark_app_setup_as_completed()`

Mark app setup as completed.

### dummy\_thumbnails.tests.helpers module

Base tests module.

- `setup_app`: Setup the app, collect the static, mark app setup as completed.

`dummy_thumbnails.tests.helpers.setup_app(collectstatic=False)`

Set up i18next.

### dummy\_thumbnails.tests.test\_core module

#### Module contents

## 10.1.2 Submodules

### 10.1.3 dummy\_thumbnails.apps module

Apps.

`class dummy_thumbnails.apps.Config(app_name, app_module)`

Bases: `django.apps.config AppConfig`

Config.

`label = 'dummy_thumbnails'`

`name = 'dummy_thumbnails'`

### 10.1.4 dummy\_thumbnails.base module

### 10.1.5 dummy\_thumbnails.conf module

Conf module.

- `get_setting`: Get app setting.

`dummy_thumbnails.conf.get_setting(setting, override=None)`

Get setting.

Get a setting from `dummy_thumbnails.conf` module, falling back to the default.

If `override` is not `None`, it will be used instead of the setting.

#### Parameters

- `setting` – String with setting name
- `override` – Value to use when no setting is available. Defaults to `None`.

**Returns** Setting value.

### 10.1.6 dummy\_thumbnails.defaults module

Defaults module.

- `IMAGES_PATH`: Path to random images.
- `VERIFY_IMAGES`: If set to `True`, images are verified.

### 10.1.7 dummy\_thumbnails.helpers module

Helpers module.

- `prepare_dirs_and_symlinks`: Prepare dirs and symlinks.

`dummy_thumbnails.helpers.prepare_dirs_and_symlinks` (*create\_dirs=False*)

Prepare dirs and symlinks.

**Parameters** `create_dirs` (*bool*) – If set to `True`, dirs are created.

### 10.1.8 dummy\_thumbnails.settings module

Settings module.

- `IMAGES_PATH`: Path to random images.
- `VERIFY_IMAGES`: If set to `True`, images are verified.

### 10.1.9 Module contents

Dummy thumbnails for most popular Django thumbnail generators.

- `default_app_config`: Default Django app config.





---

## Indices and tables

---

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



## d

- `dummy_thumbnails`, [27](#)
- `dummy_thumbnails.apps`, [26](#)
- `dummy_thumbnails.conf`, [26](#)
- `dummy_thumbnails.contrib`, [25](#)
- `dummy_thumbnails.contrib.image_importers`,  
[24](#)
- `dummy_thumbnails.contrib.thumbnailers`,  
[24](#)
- `dummy_thumbnails.contrib.thumbnailers.django_imagekit`,  
[24](#)
- `dummy_thumbnails.contrib.thumbnailers.easy_thumbnails`,  
[24](#)
- `dummy_thumbnails.contrib.thumbnailers.sorl_thumbnail`,  
[24](#)
- `dummy_thumbnails.contrib.thumbnailers.sorl_thumbnail.engines`,  
[24](#)
- `dummy_thumbnails.defaults`, [27](#)
- `dummy_thumbnails.helpers`, [27](#)
- `dummy_thumbnails.management`, [25](#)
- `dummy_thumbnails.management.commands`,  
[25](#)
- `dummy_thumbnails.management.commands.dummy_thumbnails_symlink_dummy_images`,  
[25](#)
- `dummy_thumbnails.settings`, [27](#)
- `dummy_thumbnails.tests`, [26](#)
- `dummy_thumbnails.tests.base`, [25](#)
- `dummy_thumbnails.tests.helpers`, [26](#)



## C

Command (class in `dummy_thumbnails.management.commands.dummy_thumbnails_symlink_dummy_images`), 25

Config (class in `dummy_thumbnails.apps`), 26

## D

`dummy_thumbnails` (module), 27

`dummy_thumbnails.apps` (module), 26

`dummy_thumbnails.conf` (module), 26

`dummy_thumbnails.contrib` (module), 25

`dummy_thumbnails.contrib.image_importers` (module), 24

`dummy_thumbnails.contrib.thumbnailers` (module), 24

`dummy_thumbnails.contrib.thumbnailers.django_imagekit` (module), 24

`dummy_thumbnails.contrib.thumbnailers.easy_thumbnails` (module), 24

`dummy_thumbnails.contrib.thumbnailers.sorl_thumbnail` (module), 24

`dummy_thumbnails.contrib.thumbnailers.sorl_thumbnail_engines` (module), 24

`dummy_thumbnails.defaults` (module), 27

`dummy_thumbnails.helpers` (module), 27

`dummy_thumbnails.management` (module), 25

`dummy_thumbnails.management.commands` (module), 25

`dummy_thumbnails.management.commands.dummy_thumbnails_symlink_dummy_images` (module), 25

`dummy_thumbnails.settings` (module), 27

`dummy_thumbnails.tests` (module), 26

`dummy_thumbnails.tests.base` (module), 25

`dummy_thumbnails.tests.helpers` (module), 26

## G

`get_setting()` (in module `dummy_thumbnails.conf`), 26

## H

`handle()` (`dummy_thumbnails.management.commands.dummy_thumbnails_symlink_dummy_images.Command` method), 25

`help()` (`dummy_thumbnails.management.commands.dummy_thumbnails_symlink_dummy_images.Command` attribute), 25

## I

`is_app_setup_completed()` (in module `dummy_thumbnails.tests.base`), 26

## L

`label` (`dummy_thumbnails.apps.Config` attribute), 26

`log_info()` (in module `dummy_thumbnails.tests.base`), 25

## M

`mark_app_setup_as_completed()` (in module `dummy_thumbnails.tests.base`), 26

## N

`name` (`dummy_thumbnails.apps.Config` attribute), 26

## P

`prepare_dirs_and_symlinks()` (in module `dummy_thumbnails.helpers`), 27

## S

`setup_app()` (in module `dummy_thumbnails.tests.helpers`), 26

`skip()` (in module `dummy_thumbnails.tests.base`), 25