
Cookiecutter Django CMS Documentation

Release 0.2.2

Emanuele Palazzetti

August 27, 2016

1	Basics	3
1.1	Overview	3
1.2	Usage	3
1.3	Deployment	4
2	Project info	7
2.1	Contributing	7
2.2	Authors	7
2.3	Changelog	8

Cookiecutter template to be up and running with Django CMS as fast as possible! A simple bootstrap template is available and you can use it as a base for your website.

1.1 Overview

Warning: All these steps follow the [Django CMS documentation](#) (still in beta). However this is **ONLY** a cookiecutter template and there is **NO INTENTION** to replace the official documentation.

1.1.1 Features

Features?

- Have your Django CMS installation ready with just “one command”
- Shipped with a simple Bootstrap 3 template
- Django project structure according to “Two scoops of Django” book (thanks to [@pydanny](#) and [@audreyr](#))
- Project ready to deployment on Heroku without any Django configurations
- Collect static on your Amazon S3 bucket
- Do you need anything else?

1.1.2 Similar projects

- [aldryn-installer](#) from [@nephila](#)

1.2 Usage

1.2.1 Bootstrap your project

First, get and install cookiecutter in your virtualenv:

```
$ pip install cookiecutter
```

Now run it against this repo:

```
$ cookiecutter https://github.com/palazzem/cookiecutter-django-cms.git --checkout 0.2.3
```

You'll be prompted for some project configurations:

```
full_name (default is "Michael Scott")?
email (default is "bestboss.scott@example.com")?
github_username (default is "mscott")?
year (default is "2013")?
version (default is "0.1.0")?
project_name (default is "Django CMS")?
repo_name (default is "django-cms-web")?
django_cms_app (default is "django_cms")?
project_short_description (default is "Django CMS boilerplate to start your website in 5 minutes.")?
languages (default is "en")?
site_name (default is "example.com")?
django_filer (default is "n")?
heroku (default is "y")?
```

Now you are ready to use Django CMS!

1.2.2 Initial configurations

Like any other Django project you should do these extra steps (if you are a Djangonaut, skip this).

Install all development requirements in your virtualenv:

```
$ pip install -r requirements/development.txt
```

Sync your database with migrations:

```
$ python manage.py syncdb --all --settings=django_cms.settings.dev
$ python manage.py migrate --fake --settings=django_cms.settings.dev
```

Run all Django CMS check and start django runserver:

```
$ python manage.py cms check --settings=django_cms.settings.dev
$ python manage.py runserver --settings=django_cms.settings.dev
```

Open <http://localhost:8000> and create your first page with Django CMS admin!

Note: You can avoid to use `--settings` parameter if you export `DJANGO_SETTINGS_MODULE=django_cms.settings.dev` in your environment

Note: `django_cms` package could have a different name according to your initial choose

1.2.3 More configurations

For more Django CMS configurations, check official [documentation](#) (still in beta).

1.3 Deployment

1.3.1 Prepare your git repository

Before continue, be sure to create your git repository:

```
$ git init
$ git add .
$ git commit -m"Initial commit"
```

Note: Generated project already have a valid `.gitignore` for Django

1.3.2 Heroku

If you choose to enable Heroku deployment during project bootstrap, you already have anything you need. Simply obtain [Heroku Toolbelt](#) and start creating your first application:

```
$ heroku apps:create <app_name>
$ git push heroku master
```

You have deployed your website in Heroku platform but you need to achieve some extra steps.

Note: Check `ALLOWED_HOSTS` setting or you will get a 400 (bad request) error when in production

1.3.3 Heroku configuration

Set these environment variables so production configuration will work like expected:

```
$ heroku config:set DJANGO_SECRET_KEY=<random secret key>
$ heroku config:set DJANGO_SETTINGS_MODULE=django_cms.settings.production
```

Note: `django_cms` package could have a different name according to your initial choose

Configure your [AWS bucket](#) and add these environment variables to Heroku:

```
$ heroku config:set AWS_ACCESS_KEY_ID=<random key_id>
$ heroku config:set AWS_SECRET_ACCESS_KEY=<random access_key>
$ heroku config:set AWS_STORAGE_BUCKET_NAME=<your bucket name>
```

1.3.4 Syncdb and collect static

Run these commands using Heroku run:

```
$ heroku run python django_cms/manage.py syncdb --all
$ heroku run python django_cms/manage.py migrate --fake
$ heroku run python django_cms/manage.py collectstatic
```

Note: `django_cms` package could have a different name according to your initial choose

That's all! Your Django CMS website is deployed on Heroku platform!

Project info

2.1 Contributing

Contributions are welcome, and they are greatly appreciated! Every little bit helps, and credit will always be given.

2.1.1 Add new templates

If you want to add your bootstrap template, simply submit a pull request:

- Add your template in `templates` folder
- In `base.py` add your template name in `CMS_TEMPLATES`
- Add your name in `AUTHORS.rst` under Contributors list

Note: It's better to provide generic templates that could be customized by end users. Don't put your already customized template or third party template with Copyright license.

2.1.2 More deployment options

Submit a pull request and edit all required settings to provide new deployment platforms. Be sure to:

- Add a deployment option in `cookiecutter.json`
- Use `{% if ... %}` blocks in settings files
- Add deployment instructions in `deployment.rst`

2.2 Authors

2.2.1 Development Lead

- Emanuele Palazzetti <hello@palazzetti.me>

2.2.2 Contributors

- Dacian Popute @ KuwaitNET <dacian@kuwaitnet.com>

2.3 Changelog

2.3.1 0.3 (2016-04-09)

New features:

- django compressor
- django rq

Other changes:

- Update Django CMS 3.2.3
- Update Django 1.9.5
- removed psycpg from production environment

2.3.2 0.2.3 (2013-12-23)

New features:

- Documentation with installation and usage
- Added context processor to expose Google Analytics and Debug to Django template engine

Bug fixes:

- Missing XML sitemaps on urls
- Psycpg2 is missing in production environment
- Django reversion 1.8.0 now works

Other changes:

- Update Django CMS beta3
- Update Django 1.6.1

2.3.3 0.2.2 (2013-12-07)

- Minor bootstrap updates
- Heroku deployment files
- Some bugfixes according to Django CMS beta