
Django Admin Bootstrapped3 Documentation

Release 0.6

Darryl Cousins

Nov 02, 2017

Contents

1	django-admin-bootstrapped	1
1.1	Credits	1
1.2	Requirements	1
1.3	Installation	1
1.4	Build and Run Test Project	2
1.5	Included	2

django-admin-bootstrapped

A Django admin theme using [Bootstrap](#). It doesn't need any kind of modification on your side, just add it to the installed apps.

1.1 Credits

This package is a fork of [django-admin-bootstrapped](#). All credit goes to [Riccardo Forina](#) and the other contributors. A good deal of the this readme comes from the original repository.

1.2 Requirements

- Django >=1.7.x.
- Python 3
- [django-bootstrap3](#)

1.3 Installation

Simple install into a virtualenv:

```
$ pip install git+https://github.com/darrylcousins/django-admin-bootstrapped3.git
```

Add 'django_admin_bootstrapped' into the `INSTALLED_APPS` list **before** 'django.contrib.admin'

1.4 Build and Run Test Project

Get the test project from github:

```
$ git clone https://github.com/darrylcousins/django-project.git
$ cd django-project
```

The test project uses `django-bootstrap3` and `bootstrapped3` admin. these extra packages can be installed with:

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

The test project has some tests:

```
$ python manage.py test project
```

The tables and sample data can be installed with:

```
$ python manage.py migrate
$ python manage.py loaddata project/fixtures/project.json
```

And can be run with:

```
$ python manage.py runserver 9000
```

There are no urls beyond the admin screens and api json views. It attempts to demonstrate the autocomplete widgets. All models demonstrate some aspect of the widgets.

1.5 Included

1.5.1 Autocomplete Widgets

See [django-autocomplete](#).

1.5.2 Translate/change an application name with a template

Create a file named `admin_app_name.html` into the application's template folder. Eg: `project/templates/admin_app_name.html`. You can also change the default Django Administration title, just add a `admin_title.html` file into your `project/templates/admin/` folder.

1.5.3 Add custom html to the list view or change form of any model with a template

You can inject custom html on top of any change form creating a template named `admin_model_MODELNAME_change_form.html` into the application's template folder. Likewise the title of the model list view can be changed by adding a template called `admin_model_MODELNAME_title.html` into the application's template folder.

1.5.4 Inline sortable

You can add drag&drop sorting capability to any inline with a couple of changes to your code.

First, add a `position` field in your model (and sort your model accordingly), for example:

```
class TestSortable(models.Model):
    that = models.ForeignKey(TestMe)
    position = models.PositiveSmallIntegerField("Position")
    test_char = models.CharField(max_length=5)

    class Meta:
        ordering = ('position', )
```

Then in your admin.py create a class to handle the inline using the `django_admin_bootstrapped.admin.models.SortableInline` mixin, like this:

```
from django_admin_bootstrapped.admin.models import SortableInline
from models import TestSortable

class TestSortable(admin.StackedInline, SortableInline):
    model = TestSortable
    extra = 0
```

This feature was brought to you by [Kyle Bock](#).

- search