
django-nonelfield Documentation

Release 0.4

Artur Barseghyan <artur.barseghyan@gmail.com>

Jan 23, 2020

Contents

1	Prerequisites	3
2	Installation	5
3	Usage	7
3.1	forms.py	7
3.2	serializers.py	7
4	Examples	9
5	License	11
6	Support	13
7	Author	15
8	Project documentation	17
8.1	Release history and notes	17
8.1.1	0.4	17
8.1.2	0.3	17
8.1.3	0.2	18
8.1.4	0.1	18
8.2	nonefield package	18
8.2.1	Subpackages	18
8.2.1.1	nonefield.tests package	18
8.2.1.1.1	Submodules	18
8.2.1.1.2	nonefield.tests.base module	18
8.2.1.1.3	nonefield.tests.constants module	18
8.2.1.1.4	nonefield.tests.test_nonefield module	18
8.2.1.1.5	Module contents	18
8.2.2	Submodules	18
8.2.3	nonefield.fields module	18
8.2.4	nonefield.widgets module	19
8.2.5	Module contents	19
9	Indices and tables	21
	Python Module Index	23

`django-nonefield` is a `None` field for Django.

A typical use case: allow presentational (non-input, non-field) form elements (text, image, embed video, etc). This could be very useful if your forms are dynamic (as in form-builders/generators). Django REST Framework integration is implemented as well.

CHAPTER 1

Prerequisites

- Django 1.8, 1.11, 2.0, 2.1, 2.2, 3.0.
- Python 2.7, 3.5, 3.6, 3.7, 3.8

CHAPTER 2

Installation

- (1) Install latest stable version from PyPI:

```
pip install django-nonefield
```

Or latest stable version from GitHub:

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

Or latest stable version from BitBucket:

```
pip install https://bitbucket.org/barseghyanartur/django-fobi/get/stable.tar.gz
```

- (2) Add nonefield to `INSTALLED_APPS` of the your projects' Django settings.

```
INSTALLED_APPS = (  
    # ...  
    # None field  
    'nonefield',  
    # ...  
)
```


3.1 forms.py

In forms you could use it as follows:

```
from django import forms
from nonefield.fields import NoneField

class MyForm(forms.Form):

    name = forms.CharField(max_length=255)
    some_text = NoneField(initial='Lorem ipsum')
```

See [this snippet](#) as an example of how to allow to use paragraphs in the *django-forms-builder*.

3.2 serializers.py

You can also use it in Django REST Framework.

```
from rest_framework import serializers
from nonefield.contrib.drf_integration.fields import NoneField

class ContentTextField(NoneField):
    """Content text field."""

class BarSerializer(serializers.Serializer):

    title = serializers.CharField(max_length=256)
    text = serializers.CharField()
    context_text = ContentTextField(label='', default='Haha')
```

See [how it's used in django-fobi](#) to allow to use content/presentational elements (text, image, embed video, etc.) in the Django REST Framework schema.

CHAPTER 4

Examples

- [django-forms-builder example](#)
- [django-fobi form elements example](#)
- [django-fobi DRF integration example](#)

CHAPTER 5

License

GPL-2.0-only OR LGPL-2.1-or-later

CHAPTER 6

Support

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

CHAPTER 7

Author

Artur Barseghyan <artur.barseghyan@gmail.com>

Contents:

8.1 Release history and notes

Sequence based identifiers are used for versioning (schema follows below):

`major.minor[.revision]`

- It's always safe to upgrade within the same minor version (for example, from 0.3 to 0.3.4).
- Minor version changes might be backwards incompatible. Read the release notes carefully before upgrading (for example, when upgrading from 0.3.4 to 0.4).
- All backwards incompatible changes are mentioned in this document.

8.1.1 0.4

2020-01-23

- Added Django REST Framework contrib.
- Tested against Django 2.2 and 3.0.
- Tested against Python 3.8.

8.1.2 0.3

2018-08-11

- Django 2.1 (and below) support.

8.1.3 0.2

2016-05-17

- Add `NoneField` for Django REST framework.

8.1.4 0.1

2015-06-07

- Initial release.

8.2 nonfield package

8.2.1 Subpackages

8.2.1.1 nonfield.tests package

8.2.1.1.1 Submodules

8.2.1.1.2 nonfield.tests.base module

8.2.1.1.3 nonfield.tests.constants module

8.2.1.1.4 nonfield.tests.test_nonfield module

8.2.1.1.5 Module contents

8.2.2 Submodules

8.2.3 nonfield.fields module

```
class nonfield.fields.NoneField(*, required=True, widget=None, label=None, initial=None, help_text="", error_messages=None, show_hidden_initial=False, validators=(), localize=False, disabled=False, label_suffix=None)
```

Bases: `django.forms.fields.Field`

`NoneField`.

To be used with content elements like text or images, that need to be present, for instance, in between form input elements.

bound_data (*data*, *initial*)

Bound data.

Return the value that should be shown for this field on render of a bound form, given the submitted POST data for the field and the initial data, if any.

For most fields, this will simply be data; FileFields need to handle it a bit differently.

clean (*value*)

Validate the given value and return its “cleaned” value as an appropriate Python object. Raise Validation-Error for any errors.

validate (*value*)

Validate.

Always return True (by definition).

widget

alias of *nonefield.widgets.NoneWidget*

8.2.4 nonefield.widgets module

class nonefield.widgets.**NoneWidget** (*attrs=None*)

Bases: django.forms.widgets.Widget

NoneWidget.

To be used with content elements.

media

render (*name, value, attrs=None, **kwargs*)

Render.

8.2.5 Module contents

CHAPTER 9

Indices and tables

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

n

`nonefield`, [19](#)

`nonefield.fields`, [18](#)

`nonefield.widgets`, [19](#)

B

`bound_data()` (*nonefield.fields.NoneField method*), [18](#)

C

`clean()` (*nonefield.fields.NoneField method*), [18](#)

M

`media` (*nonefield.widgets.NoneWidget attribute*), [19](#)

N

`NoneField` (*class in nonefield.fields*), [18](#)

`nonefield` (*module*), [19](#)

`nonefield.fields` (*module*), [18](#)

`nonefield.widgets` (*module*), [19](#)

`NoneWidget` (*class in nonefield.widgets*), [19](#)

R

`render()` (*nonefield.widgets.NoneWidget method*), [19](#)

V

`validate()` (*nonefield.fields.NoneField method*), [19](#)

W

`widget` (*nonefield.fields.NoneField attribute*), [19](#)