
django-specifications Documentation

Release 0.4-4-g2214047

Feinheit AG

Jan 22, 2020

Contents

1	Usage	3
2	Change log	5
2.1	Next version	5
2.2	0.4 (2019-09-13)	5
2.3	0.3 (2018-09-30)	5
2.4	0.2 (2014-10-06)	6
2.5	0.1 (2012-12-18)	6

build passing

This module offers an easy way to attach auxiliary information to Django models. It allows configuring custom forms through the administration interface.

CHAPTER 1

Usage

1. Add "specifications" to INSTALLED_APPS.
2. Create a `specification = ForeignKey("specifications.Specification")` on the model you want to use specifications with. The foreign key can be nullable or required, as you wish.
3. Create the place where the specification field data is actually stored:

```
from specifications.models import SpecificationValueFieldBase

class MyObjectField(SpecificationValueFieldBase):
    parent = models.ForeignKey(
        MyObject,
        on_delete=models.CASCADE,
        related_name="fields",
    )

    class Meta:
        ordering = ["field__group__ordering", "ordering"]
```

4. Inherit from `FormWithSpecification` when creating your `ModelForm`:

```
from specifications.forms import FormWithSpecification

class MyObjectForm(FormWithSpecification):
    class Meta:
        model = MyObject
```

5. If you want to edit models with specifications you might want to use the following snippet:

```
from specifications.admin import ModelAdminWithSpecification

class MyObjectAdmin(ModelAdminWithSpecification):
    def get_fieldsets(self, request, obj=None):
        # Define your fieldsets
        fieldsets = [...]
```

(continues on next page)

(continued from previous page)

```
if self.can_add_specification_fields(request, obj):  
    # Extend your fieldset with specification fields  
    fieldsets.extend(...)  
return fieldsets
```

The fields are available after saving a specification. The specification can be changed, but you risk losing data if you do this.

2.1 Next version

- Dropped compatibility with Python 2.

2.2 0.4 (2019-09-13)

- Added a helper for adding specification field group fieldsets.
- Fixed a bug because the code wouldn't handle the case where several specification fields were gone at once.
- Dropped compatibility guarantees with old versions of Django.

2.3 0.3 (2018-09-30)

- Reformatted the code using black.
- Added `fields = '__all__'` to the modelform as required by Django.
- Replaced the migrations generated by South.
- Added support for Django 1.8 LTS, 1.11 LTS and 2.1 while dropping support for some older versions. Added support for Python 3.
- Added a testsuite, Travis CI integration and documentation on readthedocs.io.
- Added a `ModelAdminWithSpecification` class which is especially useful if you want to group specification fields into their own fieldsets.

2.4 0.2 (2014-10-06)

- Switched from distutils to setuptools.
- Added utilities for analyzing and comparing objects with specifications.

2.5 0.1 (2012-12-18)

- Initial public version.