
feedback Documentation

Release 0.1

Ivan Gromov

October 30, 2015

1	Quick start	1
1.1	In code	1
1.2	In templates	1
2	Requriments	3
3	Settings	5
4	Customize	7
4.1	Custom templates	8
5	Mailing lists	9
6	What's new	11

Quick start

1.1 In code

Add feedback to INSTALLED_APPS

```
INSTALLED_APPS = [  
    ...  
    'feedback',  
    ...  
]
```

Example urlpatterns:

```
(r'^feedback/', include('feedback.urls'))
```

Synchronize your database models run `django syncdb`

1.2 In templates

Load tags library:

```
{% load feedback_tags %}
```

Insert template tag in your template

```
{% show_feedback [key] %},
```

where `key` is feedback form key. Default key value is `default`.

Requirements

Note that feedback uses ajax form based on jQuery, so you need to include jQuery in your page. You can install it from Google hosting:

```
<script type="text/javascript" src="http://ajax.googleapis.com/ajax/libs/jquery/1.7/jq
```

Settings

FEEDBACK_FORMS Registry for custom feedback forms. See Customize section.

FEEDBACK_FORMS_NAMES Registry for feedback form names. See Customize section.

FEEDBACK_PREFIX_KEY_FIELDS Use True if need unique id html element for form fields. Default: False.

Customize

If you want to customize default feedback form, or add your own, you have to create an application with your forms. All feedback forms should be subclasses of `feedback.forms.BaseFeedbackForm`. Here is an example of custom feedback class:

```
from django import forms
from feedback.forms import BaseFeedbackForm

class OrderForm(BaseFeedbackForm):
    name = forms.CharField(label=u'Your name', max_length=200)
    email = forms.EmailField(label=u'Your Email', max_length=200)
    phone = forms.CharField(label=u'Phone', max_length=200)
    address = forms.CharField(label=u'Your address', max_length=200)
    date = forms.CharField(label=u'Date and time', max_length=200)
    file = forms.FileField(label=u'Attach the file')

    response = forms.CharField(label=u'Comment', max_length=500,
                               widget=forms.Textarea(attrs={'cols': '30', 'rows': '5'}))
    subject = u'Custom order form'
```

`subject` attribute appears in email subject.

Then you need to put feedback forms in your settings.py:

`FEEDBACK_FORMS` - dictionary object, describes feedback forms on your site. Every key-value pair stands for feedback form object. For example:

```
FEEDBACK_FORMS = {
    'default': 'feedback.forms.FeedbackForm',
    'order': 'mysite.custom_feedback.forms.OrderForm',
}
```

You can also specify `FEEDBACK_FORMS_NAMES` option:

`FEEDBACK_FORMS_NAMES` - dictionary object, that defines the way a feedback form is displayed in Mailing list admin interface. Every key-value pair stands for feedback form object. For example:

```
FEEDBACK_FORMS_NAMES = {
    'order': 'Online order form',
}
```

Now we can include `{% show_feedback order %}` in template and get overridden form.

4.1 Custom templates

You can create custom templates if you have custom form class. Application search them by first. Template names are:

feedback/FORM_KEY/feedback.html, feedback/feedback.html for rendering form itself

feedback/FORM_KEY/thankyou.html, feedback/thankyou.html for rendering success response

feedback/FORM_KEY/email.txt, feedback/feedback_messages.txt for rendering email text

Mailing lists

You can specify a mailing list for each feedback form with admin interface. By default messages from all the feedback forms are sent to the emails specified by MANAGERS setting

What's new

0.5.0 - Compatibility with django 1.7 > 1.8. Update **ru** translation.

0.4.2 - Added prefix fields settings and changing system of receipt form class. **WARNING!** If you update version and use custom templates, change in form action on “{% url ajax_feedback key=form.get_settings_key %}”.

0.4.1 - Option to define humanized form names was added.

0.4.0 - New version. Responses in DB was deleted and mailing lists was added.