

Пользовательские настройки в Python приложениях

Igor Davydenko
2012 Kyiv.py#5

О себе

- Я - Игорь Давыденко
- Работаю в **oDesk PS**
- <http://djangonaut.blogspot.com/>
- <http://djangonaut.posterous.com/>
- <https://github.com/playpauseandstop>

Вместо предисловия

- Не создавайте велосипеды, создавайте библиотеки для ИСПОЛЬЗОВАНИЯ
- Armin Ronacher (<https://github.com/mitsuhiko>)
- Kenneth Reitz (<https://github.com/kennethreitz>)

Что такое пользовательские настройки?

- Как много раз мы добавляли `ARTICLES_PER_PAGE` или `DEFAULT_RATE` в `project.settings`?
- Что делать, если эту настройку надо поменять?

Первая идея

- Сделать так, чтобы пользователи (не разработчики) могли менять необходимые настройки!
- Настройки должны считываться один раз с базы данных между изменениями
- ВАЛИДАЦИЯ настроек!
- Только определенные пользователи должны иметь право изменять настройки

django-setman

- setman == **Settings Manager**
- <http://github.com/odeskps/django-setman>
- **БЫЛО** `from django.conf import settings`
- **Стало** `from setman import settings`
- **Настройки, доступные для редактирования определяются в**
`settings.cfg`

Configuration Definition File

settings.cfg

```
[ARTICLES_PER_PAGE]  
type = int  
default = 15
```

```
[DEFAULT_RATE]  
type = decimal  
default = 30  
min_value = 10  
max_value = 50
```

Django admin

Settings Manager

core settings

App setting
Enter any word, but not more 8 symbols length (not required).

Setting to redefine
Enter number from 0 to 64.

Project settings

Boolean
Simple checkbox field for boolean setting.

Choice
Select one of the available fruits.

Choice with labels
Check how choice setting supports choice labels.

Choice with groups
Check how choice setting supports groups.

Choice with labels and groups
Check how choice setting supports groups.

Choice with internal choices
Check how choice setting supports internal choices.

Choice with internal model choices
Check how choice setting supports internal model choices.

Choice with internal model
Check how choice setting supports internal model choices.

Плюсы

- Возможность изменения настроек администраторами сайта
- Простой интерфейс для доступа к настройкам
- Валидация настроек в UI и при работе в консоли

Минуса

- Только Django
- Только Django ORM
- Django != web + Python

Что дальше?

- Оставляем плюса
- Исправляем минуса

Поддержка framework'ов

- Django
- Flask
- Pyramid ?
- GAE ?
- `class Framework(setman.frameworks.SetmanFramework)`

Поддержка backend'ов

- Django ORM
- Filebased (ini, json, pickle)
- SQLAlchemy ?
- ...
- `class Backend(setman.backends.SetmanBackend)`

Django = django_setman + django

`settings.py`

```
INSTALLED_APPS = (  
    ...  
    'setman.frameworks.django_setman',  
    ...  
)
```

We're ready to go!

Flask = flask_setman + filebased

project/app.py

```
from flask import Flask  
from setman import settings
```

```
app = Flask(__name__)  
...  
settings.configure(  
    framework='setman.frameworks.flask_setman',  
    app=app,  
    backend='setman.backends.filebased',  
    filename='/tmp/project.json'  
)
```

We're ready to go!

Other? noframework + filebased

```
project/app.py
```

```
from setman import settings
```

```
settings.configure(  
    backend='setman.backends.filebased',  
    filename='/tmp/project.json'  
)
```

We're ready to go!

Проблемы?

- Нет проблем :)
- Все работает :)

Проблемы!

- Нет документации :(
- Мало рабочих проектов
- Flask подключается не по фен-шую
- Модульность Flask'a != хорошо для стороннего разработчика

Про модульность Flask'a

- В Django возможны настройки для приложений (`app/settings.cfg`)
- Во Flask возможны настройки для блюпринтов (`blueprint/settings.cfg`)
- Если разработчик не использует блюпринт - файлы настроек надо будет добавлять руками

wtforms? wat?

- `wtforms.fields.SelectField` не поддерживает `optgroup`
- `wtforms.validators.Required` падает для значений 0 в `IntegerField`, `FloatField`, `DecimalField`
- **Динамические поля для формы :(**

```
SettingsForm = type('SettingsForm', wtforms.form.Form, {})  
setattr(SettingsForm, 'dynamic_field', wtforms.fields.*Field())
```

Django DEMO

setman

Edit Settings

Fork me on GitHub

testapp settings

App setting

Enter any word, but not more 8 symbols length (not required).

Setting to
redefine

Enter number from 0 to 64.

Project settings

Boolean

Simple checkbox field for boolean setting.

Choice

Select one of the available fruits.

Choice with
labels

Check how choice setting supports choice labels.

Choice with
groups

Check how choice setting supports groups.

Choice with
labels and
groups

Check how choice setting supports groups.

Flask DEMO

setman

Edit Settings

Fork me on GitHub

namespace settings

Namespace
setting

Enter any word, but not more 8 symbols length (not required).

Setting to
redefine

Enter number from 0 to 64.

Project settings

Boolean

Simple checkbox field for boolean setting.

Choice

Select one of the available fruits.

Choice with
labels

Check how choice setting supports choice labels.

Choice with
groups

Check how choice setting supports groups.

Choice with
labels and
groups

Check how choice setting supports groups.

No framework demo

```
arpegius:setman playpauseandstop$ make -C testproject-noframework/ app
PYTHONPATH=... python testapp/app.py
Found 4 available setting(s)
```

```
Debug mode (None)
Enter new value, old value is True:
```

```
Number of processes (Enter number of processes which would be created by app. Between 1 and 16.)
Enter new value, old value is 4: 30
```

```
Path to hosts file (We support only *NIX systems, so please start setting with "/" symbol.)
Enter new value, old value is u'/etc/hosts':
```

```
Hourly rate (None)
Enter new value, old value is Decimal('15.00'):
```

```
Cannot save settings, error:
Ensure this value is less than or equal to 16.
```

Вопросы? Предложения?

Ах, да код

<http://github.com/playpauseandstop/setman>