Blanc Basic Events Documentation

Release 0.3

Blanc Ltd

November 09, 2015

Contents

Contents:

Overview

1.1 What is blanc-basic-events?

blanc-basic-events is a simple Django package to manage a list of recurring events and special events to be displayed on a site.

1.2 Design notes

1.2.1 Two different types of events

The events package has split events into two types - recurring events and special events. This is intentionally done to keep events as simple as possible.

1.2.2 Recurring events

Recurring events are typically weekly events which occur on a regular basis. You can pick the day of the week the event occurs on, and how frequently it occurs is up to you to describe as text - instead of being limited to a small number of options which might not describe when it occurs.

These events are only listed on the recurring events list page.

1.2.3 Special events

Special events are one-off events, although they may happen again at a future date - you'll have to add it again. This is intended for events of importance which may be promoted elsewhere on the site.

These events are listed on the events list page, and also get a page for each event.

Installation

2.1 Requirements

Before installing blanc-basic-events, you'll need a copy of Django 1.7, and blanc-basic-assets installed.

2.2 Installing blanc-basic-events

The fastest way of installing is to use pip.

Simply type:

```
pip install blanc-basic-events
```

2.2.1 Manual installation

Alternative you manually install by downloading the latest version from the blanc-basic-events page on the Python Package Index.

Download the package, unpack it and run the `setup.py` installation script:

```
python setup.py install
```

2.3 Configuring your project

Edit your Django project's settings module, ensure that the required dependencies are installed and configured, then add `blanc_basic_events` to `INSTALLED_APPS`:

```
INSTALLED_APPS = [
    ...
    'blanc_basic_assets',
    ...
    'blanc_basic_events',
]
```

Also in the settings file you should edit the title for iCal feeds:

```
EVENTS_CALENDAR_NAME = "My Site"
```

Once this is done, run `python manage.py migrate` to update your database.

Edit your Django project's URL config file, and add the URL pattern for events:

```
urlpatterns = [
    ...

    # Events
    url(r'^events/', include('blanc_basic_events.urls', namespace='blanc_basic_events')),
]
```

Then your project will be ready to use the events package.

Settings

3.1 EVENTS_CALENDAR_NAME

Default: `'Events'`

The name of the calendar used for the iCal feed. You'll probably want to replace this with the name of the site.

3.2 EVENTS_CALENDAR_DESCRIPTION

Default: `'Events Calendar'`

The long description for the name of the calendar.

3.3 EVENTS_START_SUNDAY

Default: `True`

For recurring events, this will change the choices for the day of the week field. The default is `True` - making Sunday the start of the week. Setting this to `False` will make Monday the start of the week.

Note: Changing this value after deployment will not change any existing data. Sunday at the start of the week is stored as `0`, and Sunday at the end of the week is stored as `7`.

Your database will need updating after changing this value.