Blanc Basic Assets Documentation

Release 0.3

Blanc Ltd

December 04, 2015

Contents

1 Overview **3**
- 1.1 What is blanc-basic-assets? . . . 3
- 1.2 Design notes . . . 3

2 Installation **5**
- 2.1 Requirements . . . 5
- 2.2 Installing blanc-basic-assets . . . 5
- 2.3 Configuring your project . . . 5

3 Usage **7**
- 3.1 Models . . . 7

Contents:

CHAPTER 1

Overview

1.1 What is blanc-basic-assets?

blanc-basic-assets is a simple Django package to allow image and file uploads to the Django admin.

Assets are a useful place to put files or images which are referenced in other places of a site which can't justify their own image or file field, such as WYSIWYG content boxes which are used in models.

Assets can also be used as a `ForeignKey` for models where the same image or file might be used multiple times. Instead of forcing a user to upload the same image each time for the same model, it allows site managers to reuse the same asset through the entire site.

1.2 Design notes

Compared to typical Django file and image upload fields, there are two main differences which users should be aware of.

1.2.1 Deleted assets deletes the file

If you delete the model for an image or file asset, the file which has been uploaded will be deleted as well. Django usually keeps uploaded files by default, however to keep the site clean of cruft any deleted file or image objects will be deleted from the filesystem.

1.2.2 New uploads of a file deletes the old file

Django usually keeps the original version of the file intact, and any file uploads with the same name will be changed, so future uploads of example.txt will be example-1.txt, example-2.txt, etc.

To try and keep the original filename, the old file associated with an asset will be deleted - allowing the new version of a file to keep the same name. This is useful when you want to upload a new version of a file referenced from a link in a WYSIWYG.

CHAPTER 2

Installation

2.1 Requirements

Before installing blanc-basic-assets, you'll need a copy of Django 1.7 or later installed. Image uploads require the Pillow library installed.

2.2 Installing blanc-basic-assets

The fastest way of installing is to use pip.

Simply type:

```
pip install blanc-basic-assets
```

2.2.1 Manual installation

Alternative you manually install by downloading the latest version from the blanc-basic-assets 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, and add `blanc_basic_assets`:

```
INSTALLED_APPS = (
    ...
    'blanc_basic_assets',
)
```

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

CHAPTER 3

Usage

3.1 Models

All file and image assets are grouped by category, however by default the Django admin and any other Django forms will only display `ForeignKey` fields as a single list. To make asset selection easier in forms, use the `AssetForeignKey` field instead as a drop-in replacement for `ForeignKey`.

For images:

```
from django.db import models
from blanc_basic_assets.fields import AssetForeignKey

class Album(models.Model):
    title = models.CharField(max_length=100)
    image = AssetForeignKey('assets.Image')
```

For files, an optional example:

```
from django.db import models
from blanc_basic_assets.fields import AssetForeignKey

class Post(models.Model):
    title = models.CharField(max_length=100)
    attachment = AssetForeignKey('assets.File', null=True, blank=True)
```