
django-embed-video Documentation

Release 0.3

Juda Kaleta

August 22, 2013

CONTENTS

Django app for easy embedding YouTube and Vimeo videos and music from SoundCloud.

Repository is located on GitHub: <https://github.com/yetty/django-embed-video>

INSTALLATION & SETUP

1.1 Installation

Use pip to install package:

```
pip install django-embed-video
```

If you want latest version, you may use Git. It is fresh, but unstable.

```
pip install git+https://github.com/yetty/django-embed-video.git
```

1.2 Setup

Add `embed_video` to `INSTALLED_APPS` in your Django settings.

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


EXAMPLES

2.1 Template examples

First you have to load the *embed_video_tags* template tags in your template:

```
{% load embed_video_tags %}
```

Simple embedding of video:

```
{% video item.video as my_video %}
  {{ my_video|embed:'small' }}
{% endvideo %}
```

Default sizes are `tiny` (420x315), `small` (480x360), `medium` (640x480), `large` (960x720) and `huge` (1280x960). You can set your own size:

```
{{ my_video|embed:'800x600' }}
```

Usage of variables:

```
{% video item.video as my_video %}
  URL: {{ my_video.url }}
  Thumbnail: {{ my_video.thumbnail }}
  Backend: {{ my_video.backend }}
{% endvideo %}
```

There is a simpler way, if you don't need work with parameters as `my_video.url` or `my_video.thumbnail` and you want to use just `embed` tag.

```
{{ 'http://www.youtube.com/watch?v=guXyvo2FfIs'|embed:'large' }}
```

2.2 Model examples

Using the `EmbedVideoField` provides you validation of URLs.

```
from django.db import models
from embed_video.fields import EmbedVideoField

class Item(models.Model):
    video = EmbedVideoField() # same like models.URLField()
```

2.3 Admin examples

Todo

Admin mixins are on TODO list. Be patient or send *[pull request](#)* :).

2.4 Custom backends

If you have specific needs and default backends don't suits you, you can write your custom backend.

my_project/my_app/backends.py:

```
from embed_video.backends import VideoBackend

class CustomBackend(VideoBackend):
    re_detect = re.compile(r'http://myvideo\.com/[0-9]+')
    re_code = re.compile(r'http://myvideo\.com/(?P<code>[0-9]+)')

    pattern_url = 'http://play.myvideo.com/c/%s/'
    pattern_thumbnail_url = 'http://thumb.myvideo.com/c/%s/'
```

You can also overwrite `VideoBackend` methods, if using regular expressions isn't good enough for you.

my_project/my_project/settings.py:

```
EMBED_VIDEO_BACKENDS = (
    'embed_video.backends.YoutubeBackend',
    'embed_video.backends.VimeoBackend',
    'embed_video.backends.SoundCloudBackend',
    'my_app.backends.CustomBackend',
)
```

2.5 Low level API examples

You can get instance of `VideoBackend` in your python code thanks to `detect_backend()`:

```
from embed_video.backends import detect_backend

my_video = detect_backend('http://www.youtube.com/watch?v=H4tAOexHdR4')
```

DEVELOPMENT

3.1 Contributing

I will be really pleased if you will provide patch to this Django app. Feel free in changing source code, but please keep PEP8 rules and Zen.

It is a good habit to cover your patches with *tests*.

Our repository is placed on this URL: <https://github.com/yetty/django-embed-video>

3.2 Testing

3.2.1 Requirements

You need `nose`, `mock` and `south` libraries to run tests.

```
pip install nose
pip install mock
pip install south
```

3.2.2 Running tests

Run tests with this command:

```
nosetests
```

Be sure to run it before each commit and fix broken tests.

Run tests with coverage:

```
nosetests --with-coverage --cover-package=embed_video
```

3.3 Changelog

3.3.1 Release 0.4 (Aug. 22, 2013)

- Documentation was rewritten and moved to <http://django-embed-video.rtfld.org/> .

- Custom backends (<http://django-embed-video.rtfld.org/en/latest/examples.html#custom-backends>).
- Improved YouTube and Vimeo regex.
- Support for Python 3.
- Renamed base to backends.

3.3.2 Release 0.3 (Aug. 20, 2013)

- Security fix: faked urls are treated as invalid. See [this page](#) for more details.
- Fixes:
 - allow of empty video field.
 - requirements in setup.py
- Added simpler way to embed video in one-line template tag:

```
{{ 'http://www.youtube.com/watch?v=guXyvo2FfLs' | embed:'large' }}
```

Usage:

```
{% video item.video as my_video %}
    Backend: {{ my_video.backend }}
{% endvideo %}
```

3.3.3 Release 0.2 (June 25, 2013)

- Support of SoundCloud

3.3.4 Release 0.1 (June 1, 2013)

- Initial release

3.4 TODOs list

- provide AdminEmbedVideoMixin
- Vimeo thumbnail

Todo

Admin mixins are on TODO list. Be patient or send *[pull request](#)* :).

(The *[original entry](#)* is located in /var/build/user_builds/django-embed-video/checkouts/v0.4/docs/examples.rst, line 72.)

WEBSITES USING DJANGO-EMBED-VIDEO

- [Tchorici.cz \(sources\)](#)

Are you using `django-embed-video`? Send pull request!

API

5.1 API reference

5.1.1 Backends

class `embed_video.backends.SoundCloudBackend(url)`
Backend for SoundCloud URLs.

class `embed_video.backends.VideoBackend(url)`
Base backend, good to inherit.

get_code()
Returns searched code from URL by `re_code`.

get_thumbnail_url()
Returns thumbnail URL folded from `pattern_thumbnail_url` and parsed code.

get_url()
Returns URL folded from `pattern_url` and parsed code.

classmethod is_valid(klass, url)
Class method to control if passed url is valid for current backend. By default it is done by `re_detect` regex.

pattern_thumbnail_url = None
Pattern in which the code is inserted to get thumbnail url.
Example: `http://static.myvideo.com/thumbs/%s`

pattern_url = None
Pattern in which the code is inserted.
Example: `http://myvideo.com?code=%s`

re_code = None
Compiled regex (`re.compile()`) to search code in URL.
Example: `re.compile(r'myvideo\.com/\?code=(?P<code>\w+)')`

re_detect = None
Compilede regex (`re.compile()`) to detect, if input URL is valid for current backend.
Example: `re.compile(r'^http://myvideo\.com/.*)`

class `embed_video.backends.VimeoBackend(url)`
Backend for Vimeo URLs.

class `embed_video.backends.YoutubeBackend(url)`
Backend for YouTube URLs.

`embed_video.backends.detect_backend(url)`
Detect the right backend for given URL.

5.1.2 Fields

class `embed_video.fields.EmbedVideoField(verbose_name=None, name=None, **kwargs)`
Model field for embedded video. Descendant of `django.db.models.URLField`.

class `embed_video.fields.EmbedVideoFormField(max_length=None, min_length=None, *args, **kwargs)`
Form field for embedded video. Descendant of `django.forms.URLField`

5.1.3 Template tags

You have to load template tag library first.

```
{% load embed_video_tags %}
```

class `embed_video.templatetags.embed_video_tags.VideoNode(parser, token)`
Template tag video. It gives access to all `VideoBackend` variables.

Usage:

```
{% video URL as VAR %}
...
{% endvideo %}
```

Example:

```
{% video item.video as my_video %}
URL: {{ my_video.url }}
Thumbnail: {{ my_video.thumbnail }}
Backend: {{ my_video.backend }}
{% endvideo %}
```

`embed_video.templatetags.embed_video_tags.embed(backend, size='small')`
Shortcut for `VideoNode()` tag.

Usage:

```
{{ URL|embed:SIZE }}
```

Example:

```
{{ 'http://www.youtube.com/watch?v=guXyvo2FfLs' |embed:'large' }}
```

Predefined sizes:

size	width	height
tiny	420	315
small	480	360
medium	640	480
large	960	720
huge	1280	960

You can also use custom size - in format WIDTHxHEIGHT (eg. 500x400).

5.1.4 Utils

`embed_video.utils.import_by_path(dotted_path, error_prefix='')`

Import a dotted module path and return the attribute/class designated by the last name in the path. Raise `ImportError` if something goes wrong.

Warning: Deprecated since version Django: 1.6 Function `django.utils.module_loading.import_by_path()` has been added in Django 1.6.

INDICES AND TABLES

- *genindex*
- *modindex*
- *search*

PYTHON MODULE INDEX

e

`embed_video.backends, ??`
`embed_video.fields, ??`
`embed_video.templatetags.embed_video_tags,`
`??`
`embed_video.utils, ??`