django-photologue Documentation

Release 2.4

Justin Driscoll

May 15, 2013

CONTENTS

The Photologue documentation is being transferred from a Google Code wiki to a Sphinx-generated setup (if you're reading this at readthedocs.org, then you're looking at these docs).

This documentation is accurate and up-to-date (hopefully!); the old docs on the Google Code wiki are extensive but have not been updated in a long time.

Contents:

CHAPTER

ONE

INSTALLATION

1.1 Introduction

The easiest way to install Photologue is with pip:

```
pip install django-photologue
```

You can verify Photologue is available to your project by running the following commands from within your project directory:

```
manage.py shell
>>> import photologue
>>> photologue.VERSION
(2, 0, 'rc1')
```

1.2 Dependencies

- PIL.
- South.

These 2 apps will be installed automatically if they are not already there.

Photologue also uses the Django admin app, so enable it if you have not already done so.

1.3 Configure Your Django Settings

1. Add 'photologue' to your INSTALLED_APPS setting:

   ```
   INSTALLED_APPS = (
        # ...other installed applications,
        'photologue',
        'south',
   )
   ```

2. Confirm that your MEDIA_ROOT and MEDIA_URL settings are correct (Photologue will store uploaded files in a folder called 'photologue' under your MEDIA_ROOT).

1.4 Add the urls

Add photologue to your projects urls.py file:

```
urlpatterns += patterns('',
    ...
    (r'^photologue/', include('photologue.urls')),
)
```

1.5 Sync Your Database

Use South to setup the new tables:

```
python manage.py migrate photologue
```

After the database in initialized, run the following command to setup some default values for Photologue:

```
python manage.py plinit
```

1.6 Instant Photo Gallery

Photologue comes with basic templates for galleries and photos. You can of course override them, or completely replace them. Note that all Photologue templates inherit from `photologue/root.html`, which itself just inherits from a site-wide `base.html` - you can change this to use a different base template.

CHAPTER

TWO

CONTRIBUTING TO PHOTOLOGUE

Contributions are always very welcome. Even if you have never contributed to an open-source project before - please do not hesitate to offer help. Fixes for typos in the documentation, extra unit tests, etc... are welcome. And look in the issues list for anything tagged “easy_win”.

2.1 Workflow

Django-photologue is hosted on Github, so if you have not already done so, read the excellent Github help pages. We try to keep the workflow as simple as possible, so we more-or-less follow the recommendations in the “GitHub Flow” blog post.

- The “more or less” is because we don’t do immediate releases.

2.2 Coding style

Nothing surprising here - just try to follow the conventions used by Django itself.

2.3 New features

If you’re interested in developing a new feature for Photologue, it is recommended that you first discuss it on the mailing list so as not to do any work that might not get merged in anyway.

2.4 Unit tests

Including unit tests with your contributions will earn you bonus points, maybe even a beer. So write plenty of tests.

2.5 Documentation

Keeping the documentation up-to-date is very important - so if your code changes how Photologue works, please check that the documentation is still accurate, and update it if required.

We use Sphinx to prepare the documentation; please refer to the excellent docs on that site for help.

P.S. The CHANGELOG is part of the documentation :-) so if your patch needs the end user to do something - e.g. run a South migration - don't forget to update it!

2.6 Translations

Photologue manages the application translations with Transifex. Contributions are very welcome, either by editing the translations directly on the Transifex site, or by submitting pull requests with updated .po files.

2.7 Finally

Remember that the maintainer looks after django-photologue in his spare time - so it might be a few weeks before your pull request gets looked at... and the pull requests that are nicely formatted, with code, tests and docs included, will always get reviewed first :-)

CHAPTER THREE

INDICES AND TABLES

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