django-dress-blog Documentation

Release 0.1a1

Daniel Rus Morales

Mar 20, 2017

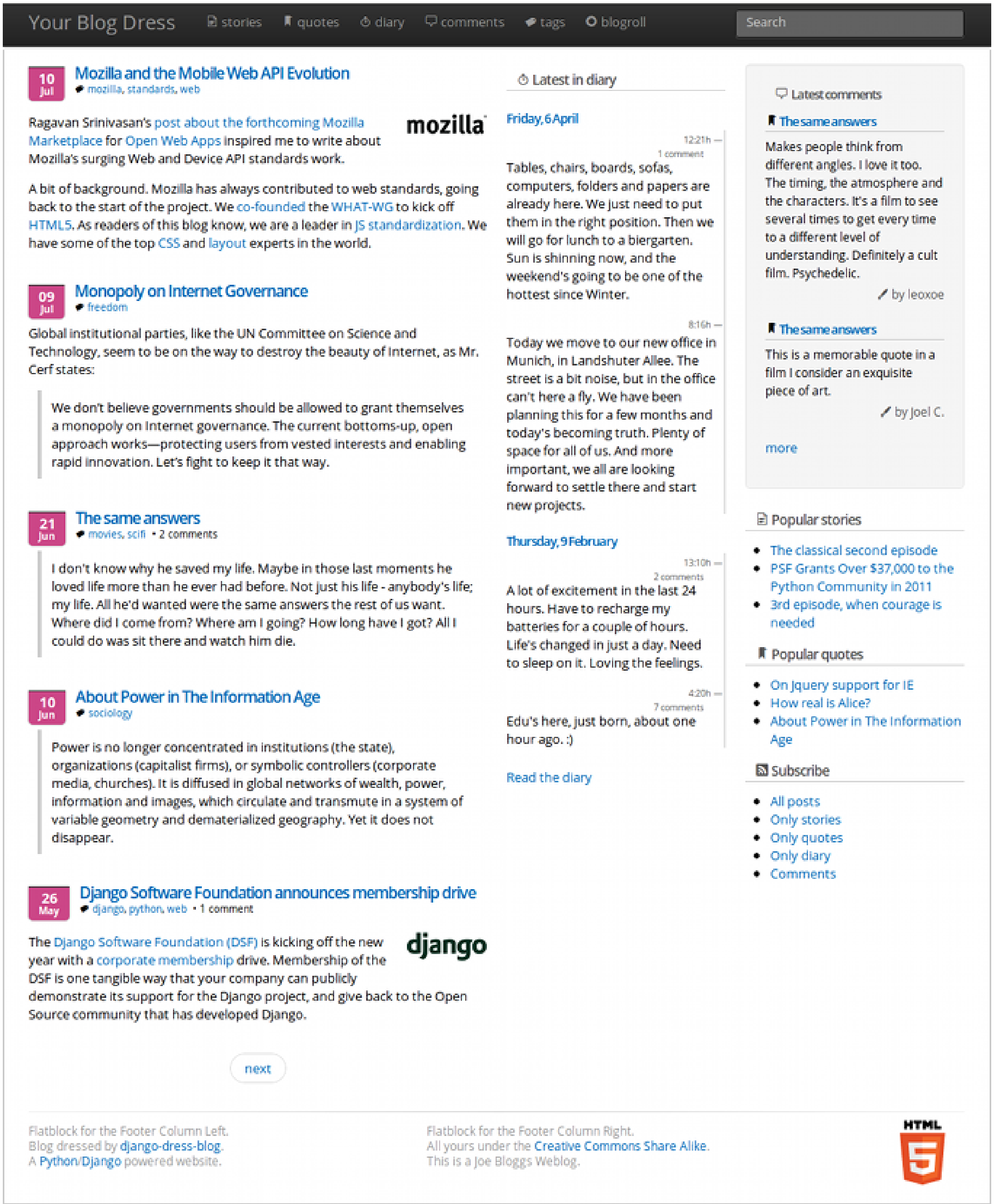
Contents

A pluggable application that provides blog capabilities to your Django project.

Documentation work in progress!

Features:

1. Three different type of posts: Stories, Quotes and Diary Entries.
2. Choose between either 3 or 4 columns layout format.
3. Default theme based on twitter-bootstrap.
4. Create new themes without modifying existing code.
5. Posts may be in Draft/Public status, and published in the future.
6. Posts categorized with tags, with django-tagging.
7. Independent paginated list of Stories, Quotes and Comments.
8. Comments managed with django-comments-xtd.
9. Support for inline media with django-inline-media.
10. Blogroll, Multiple Authors, Search capabilities with django-haystack, and more.



10 Jul **Mozilla and the Mobile Web API Evolution**
mozilla, standards, web

Ragavan Srinivasan's post about the forthcoming Mozilla Marketplace for Open Web Apps inspired me to write about Mozilla's surging Web and Device API standards work.

A bit of background. Mozilla has always contributed to web standards, going back to the start of the project. We co-founded the WHAT-WG to kick off HTML5. As readers of this blog know, we are a leader in JS standardization. We have some of the top CSS and layout experts in the world.

09 Jul **Monopoly on Internet Governance**
freedom

Global institutional parties, like the UN Committee on Science and Technology, seem to be on the way to destroy the beauty of Internet, as Mr. Cerf states:

> We don't believe governments should be allowed to grant themselves a monopoly on Internet governance. The current bottoms-up, open approach works—protecting users from vested interests and enabling rapid innovation. Let's fight to keep it that way.

21 Jun **The same answers**
movies, scifi • 2 comments

> I don't know why he saved my life. Maybe in those last moments he loved life more than he ever had before. Not just his life - anybody's life; my life. All he'd wanted were the same answers the rest of us want. Where did I come from? Where am I going? How long have I got? All I could do was sit there and watch him die.

10 Jun **About Power in The Information Age**
sociology

> Power is no longer concentrated in institutions (the state), organizations (capitalist firms), or symbolic controllers (corporate media, churches). It is diffused in global networks of wealth, power, information and images, which circulate and transmute in a system of variable geometry and dematerialized geography. Yet it does not disappear.

26 May **Django Software Foundation announces membership drive**
django, python, web • 1 comment

The Django Software Foundation (DSF) is kicking off the new year with a corporate membership drive. Membership of the DSF is one tangible way that your company can publicly demonstrate its support for the Django project, and give back to the Open Source community that has developed Django.

next

Latest in diary

Friday, 6 April

12:21h —
1 comment

Tables, chairs, boards, sofas, computers, folders and papers are already here. We just need to put them in the right position. Then we will go for lunch to a biergarten. Sun is shinning now, and the weekend's going to be one of the hottest since Winter.

8:16h —

Today we move to our new office in Munich, in Landshuter Allee. The street is a bit noise, but in the office can't here a fly. We have been planning this for a few months and today's becoming truth. Plenty of space for all of us. And more important, we all are looking forward to settle there and start new projects.

Thursday, 9 February

13:10h —
2 comments

A lot of excitement in the last 24 hours. Have to recharge my batteries for a couple of hours. Life's changed in just a day. Need to sleep on it. Loving the feelings.

4:20h —
7 comments

Edu's here, just born, about one hour ago. :)

Read the diary

Latest comments

The same answers

Makes people think from different angles. I love it too. The timing, the atmosphere and the characters. It's a film to see several times to get every time to a different level of understanding. Definitely a cult film. Psychedelic.

by leoxoe

The same answers

This is a memorable quote in a film I consider an exquisite piece of art.

by Joel C.

more

Popular stories

- The classical second episode
- PSF Grants Over $37,000 to the Python Community in 2011
- 3rd episode, when courage is needed

Popular quotes

- On Jquery support for IE
- How real is Alice?
- About Power in The Information Age

Subscribe

- All posts
- Only stories
- Only quotes
- Only diary
- Comments

Flatblock for the Footer Column Left.
Blog dressed by django-dress-blog.
A Python/Django powered website.

Flatblock for the Footer Column Right.
All yours under the Creative Commons Share Alike.
This is a Joe Bloggs Weblog.

Run the demo project to see django-dress-blog in action.

Contents:

CHAPTER 1

Tutorial

Django-dress-blog is a blogging app that allows authors post stories, quotes and diary entries.

Motivation

Django-dress-blog is a blogging app inspired by Kevin Fricovsky's django-mingus that offers authors three different type of posts: **stories** to write their own content, **quotes** to cite other people's content, and the **diary** to micro-post on anything in a quick fashion.

Django-dress-blog is suitable for start-ups and other collaborative projects, may show posts' author, and it comes with a fully functional theme based on twitter-bootstrap and support for extra themes.

Its search functionality make use of django-haystack, which supports up to four different search engines. The default installation uses xapian-haystack.

Using Django-dress-blog authors categorize content by tagging, what produces a Folksonomy, allowing categorization on the go rather than enforcing category creation in advance.

Installation

Installing Django-dress-blog is as simple as checking out the sources and adding the path to your project or `PYTHONPATH`.

Use git, pip or easy_install to check out Django-dress-blog from Github or get a release from PyPI:

1. Use **git** to clone the repository, and then install the package (read more about git):

- `git clone git://github.com/danirus/django-dress-blog.git` and
- `python setup.py install`

2. Or use **pip** (read more about pip):

- Do `pip install django-dress-blog`, or

- Edit your project's `requirements` file and append either the Github URL or the package name `django-dress-blog`, and then do `pip install -r requirements`.

3. Or use **easy_install** (read more about easy_install):

- Do `easy_install django-dress-blog`

Configuration

It is assumed you have pip installed and virtualenv. Although virtualenv is not a requirement, just a healthy recommendation. Configuration comprehends the following steps:

1. Install the dependencies listed in the `requirements` file:

```
pip install -r requirements
```

2. Add the following entries to `INSTALLED_APPS` in your `settings.py` file:

```
'django.contrib.staticfiles',
'django.contrib.comments',
'django_comments_xtd',
'django_markup',
'inline_media',
'flatblocks',
'sorl.thumbnail',
'tagging',
'request',
'dress_blog',
'haystack',
```

3. Some of the new installed apps requires extra configuration. Edit your `settings.py` file again and append the next entries:

- For the comments app:

```
COMMENTS_APP = 'django_comments_xtd'
COMMENTS_XTD_CONFIRM_EMAIL = True
COMMENTS_XTD_SALT = 'es-war-einmal-una-bella-princesa-in-a-beautiful-castle'
```

- For Django-tagging:

```
FORCE_LOWERCASE_TAGS = True
```

- For sorl.thumbnail:

```
THUMBNAIL_BACKEND = 'inline_media.sorl_backends.AutoFormatBackend'
THUMBNAIL_FORMAT = 'JPEG'
```

- For request:

```
REQUEST_IGNORE_AJAX = True
REQUEST_IGNORE_USERNAME = ["admin"]
REQUEST_IGNORE_PATHS = (r'^admin/', r'^favicon.ico')
REQUEST_TRAFFIC_MODULES = (
    'request.traffic.UniqueVisitor',
    'request.traffic.UniqueVisit',
    'request.traffic.Hit',
    'request.traffic.Search',
)
```

- For Django-haystack, when using the Xapian search engine:

```
HAYSTACK_CONNECTIONS = {
    'default': {
        'ENGINE': 'haystack.backends.xapian_backend.XapianEngine',
        'PATH': os.path.join(os.path.dirname(__file__), 'xapian_index'),
    },
}
```

- For Django-dress-blog:

```
DRESS_BLOG_PAGINATE_BY = 10
DRESS_BLOG_UI_COLUMNS = 3
DRESS_BLOG_THEMES_PATH = os.path.join(STATIC_ROOT, 'dress_blog/themes')
```

4. To use the XapianEngine backend with Haystack you have to copy the `xapian_backend.py` file from the xapian_haystack directory into django-haystack's backends directory. Go to your VirtualEnv `src/` directory and copy the file:

```
cp xapian-haystack/xapian_backend.py django-haystack/haystack/backends/
```

5. Add a new context processor to `TEMPLATE_CONTEXT_PROCESSORS` in your `settings.py` file:

```
TEMPLATE_CONTEXT_PROCESSORS = (
    ...
    'dress_blog.context_processors.config',
)
```

6. Edit your `urls.py` file and add the following entries:

```
url(r'^blog/',            include('dress_blog.urls')),
url(r"^comments/",        include("django_comments_xtd.urls")),
```

7. Collect static files provided with Django-dress-blog:

```
python manage.py collectstatic
```

8. Build the Xapian search index in a regular basis to track new content added to your blog by running the following command:

```
python manage.py rebuild_index
```

9. Create database data structures:

```
python manage.py syncdb
```

10. Run you project dev web server and hit your project's dress-blog URL <http://localhost:8000/blog>_:

    python manage.py runserver

CHAPTER 2

Example

CHAPTER 3

Theming

CHAPTER 4

Templatetags

CHAPTER 5

Settings

CHAPTER 6

Indices and tables

- genindex
- modindex
- search