
django-layar Documentation

Release 0.2.0

James Turk

January 05, 2017

1	django-layar	3
1.1	Requirements	3
1.2	Usage	3
2	Module Documentation	5
3	Changelog	7
3.1	0.2.0 - March 29 2010	7
3.2	0.1.1 - March 4 2010	7
3.3	0.1.0 - October 22 2009	7
4	License	9
5	Indices and tables	11

Contents:

django-layar

Django generic view for making [Layar](#) endpoints.

Provides abstract class that responds to Layar API requests in the appropriate format. By implementing two small functions it is possible to add a layer to the Layar augmented reality application for Android and iPhone.

django-layar is a project of Sunlight Labs (c) 2009. Written by James Turk <jturk@sunlightfoundation.com>

Source: <http://github.com/sunlightlabs/django-layar/>

Documentation: <http://packages.python.org/django-layar/README.html>

1.1 Requirements

python >= 2.4

django >= 1.0

1.2 Usage

1.2.1 Layar Developer Key

If you haven't already, it is necessary to [sign up with Layar](#) to obtain an API key.

Set `LAYAR_DEVELOPER_KEY='<YOUR_KEY>'` in `settings.py`

1.2.2 Creating a `LayarView` subclass

django-layar provides a class-based generic view at `layar.LayarView`. In order to provide your own layers it is necessary to inherit from this view and implement two simple functions (per layer).

The required functions are `get_LAYERNAME_queryset()` and `poi_from_LAYERNAME_item()`.

It is possible to serve as many layers as you desire from a single Layar endpoint, just make sure that the name of your functions matches the name you provide when registering your layers with Layar.com

`get_LAYERNAME_queryset()` is passed a number of arguments (see `layar.LayarView` for detail) and should return a queryset of the objects for `LayarView` to paginate and return.

`poi_from_LAYERNAME_item()` is called on each item being returned, and should convert items into `layar.POI` objects.

It is usually best to then create an instance of your derived class in your application's `views.py`

Example:

```
# views.py

from django.contrib.gis.geos import Point
from myapp.models import BusStop
from layar import LayarView, POI

class BusStopLayar(LayarView):

    # make sure to accept **kwargs
    def get_busstop_queryset(self, latitude, longitude, radius, **kwargs):
        return BusStop.objects.filter(location__distance_lt=(Point(longitude, latitude), radius))

    def poi_from_recoverygov_item(self, item):
        return POI(id=item.id, lat=item.location.y, lon=item.location.x, title=item.name,
                  line2=item.route_name, line3='Distance: %distance%')

# create an instance of BusStopLayar
busstop_layar = BusStopLayar()
```

In `urls.py` it is then necessary to map a URL directly to `busstop_layar`:

```
# urls.py

from django.conf.urls.defaults import *

urlpatterns = patterns('',
    url(r'^layar_endpoint/$', 'myapp.views.busstop_layar'),
)
```

Module Documentation

Changelog

3.1 0.2.0 - March 29 2010

- new Laya3 parameters: alt, search2/3, slider2/3, checkboxes
- made accuracy and radius optional
- add default radius handling

3.2 0.1.1 - March 4 2010

- fix packaging issues
- lat/lng conversion on Decimal objects
- warn when POI.actions is a dict

3.3 0.1.0 - October 22 2009

- initial working release

License

Copyright (c) 2009, Sunlight Foundation All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the name of Sunlight Foundation nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Indices and tables

- `genindex`
- `modindex`
- `search`