Geo Web applications: GeoDjango

Option GIS-Python

hes·so business.

Jean-Paul Calbimonte

School of Management

Bachelor of Science HES-SO (BSc) in Business Information Technology

swissuniversities

What is GeoDjango?

https://docs.djangoproject.com/en/3.0/ref/contrib/gis/

- Contrib module for Django
- Geographic Web framework for Django
- Simple as possible to create geographic Web applications
- Django model fields for OGC geometries and raster data.
- Extensions to Django's ORM for querying and manipulating spatial data.
- Loosely-coupled, high-level Python interfaces for GIS geometry and raster operations
- Data manipulation in different formats.
- Editing geometry fields from the admin.

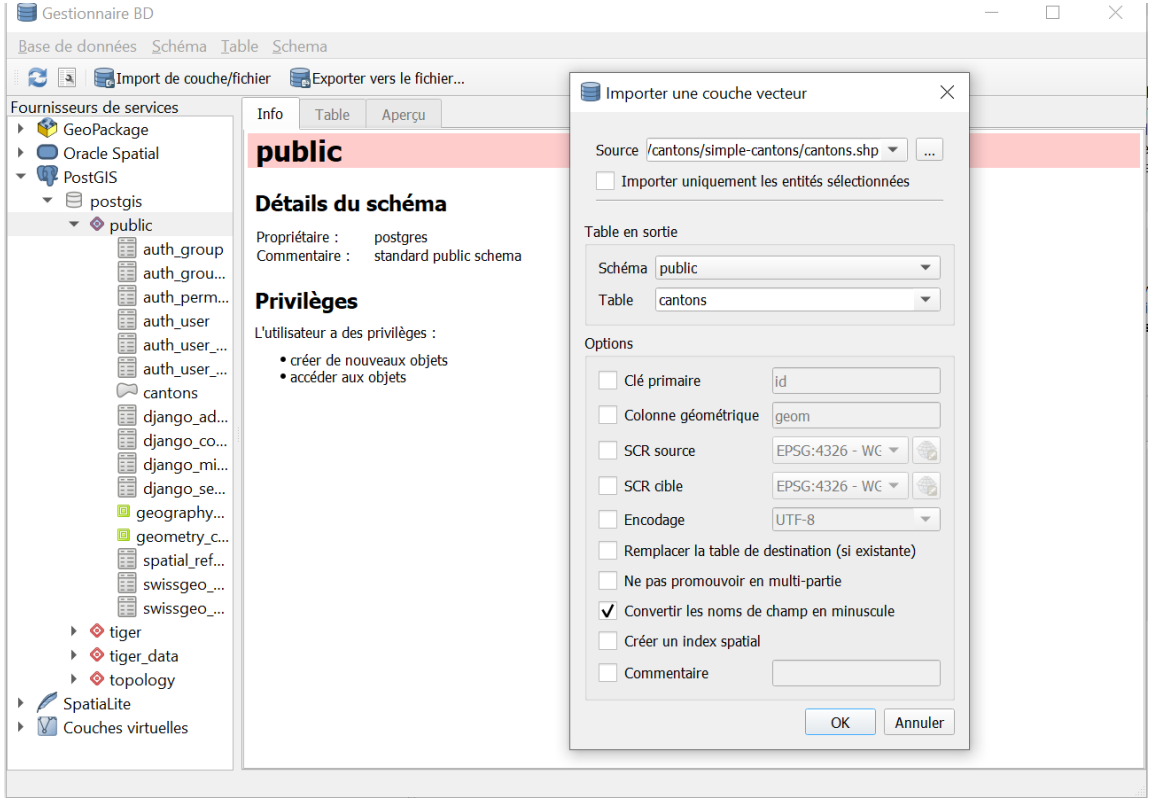
> settings.py

```
INSTALLED_APPS = [',
 'django.contrib.admin',
 'django.contrib.auth',
 'django.contrib.contenttypes',
 'django.contrib.sessions',
 'django.contrib.messages',
 'django.contrib.staticfiles',
 'django.contrib.gis',
]
```

```
DATABASES = {
 'default': {
   'ENGINE': 'django.contrib.gis.db.backends.postgis',
   'NAME': 'postgres',
      'USER': 'postgres',
      'PASSWORD': 'admin',
      'HOST': 'db',
      'PORT': '5432',  }
}
```

> Import cantons shapefile with QGIS

Get the file at: data/cantons/simple-cantons/cantons.shp
Go to QGIS>Base de données>Gestionnaire BD



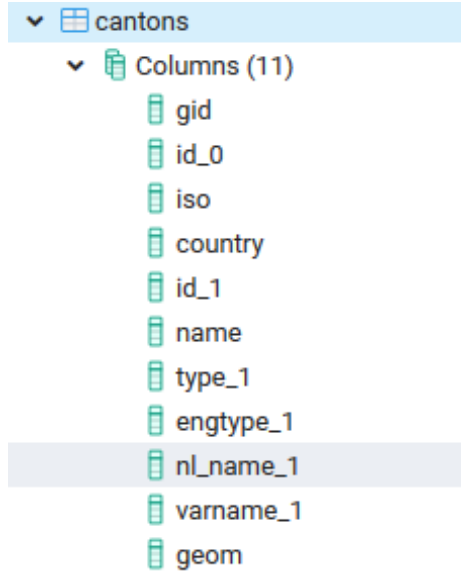
> models.py

```
from django.contrib.gis.db import models
```

```
class Canton(models.Model):
    id=models.PositiveIntegerField(primary_key=True)
    name=models.CharField(max_length=200)
    geom=models.MultiPolygonField(srid=4326,null=True)

    class Meta:
        db_table = "cantons"

    def __str__(self):
        return self.name
```

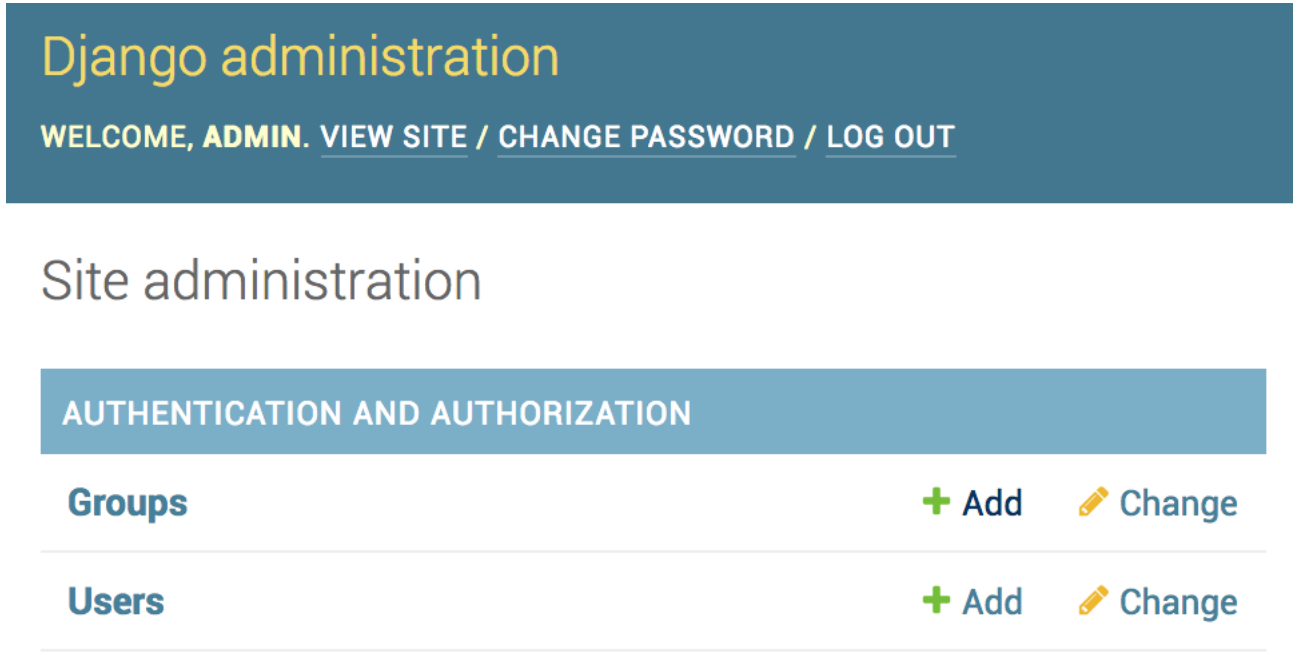


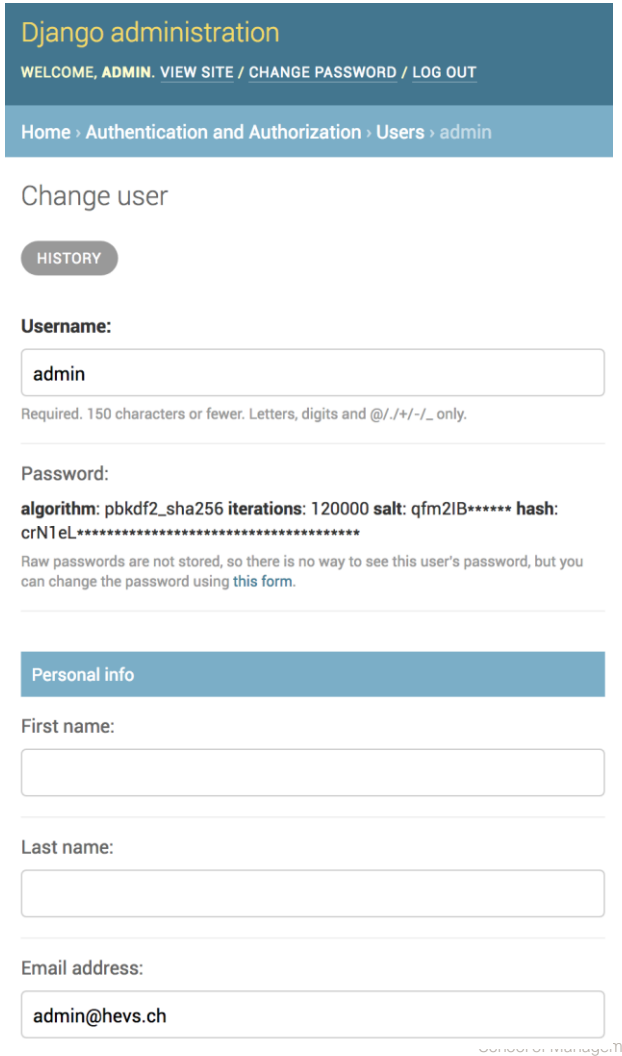

```
> python manage.py makemigrations swissgeo
```

```
> python manage.py migrate  --fake
```

> Django admin

http://127.0.0.1:8000/admin

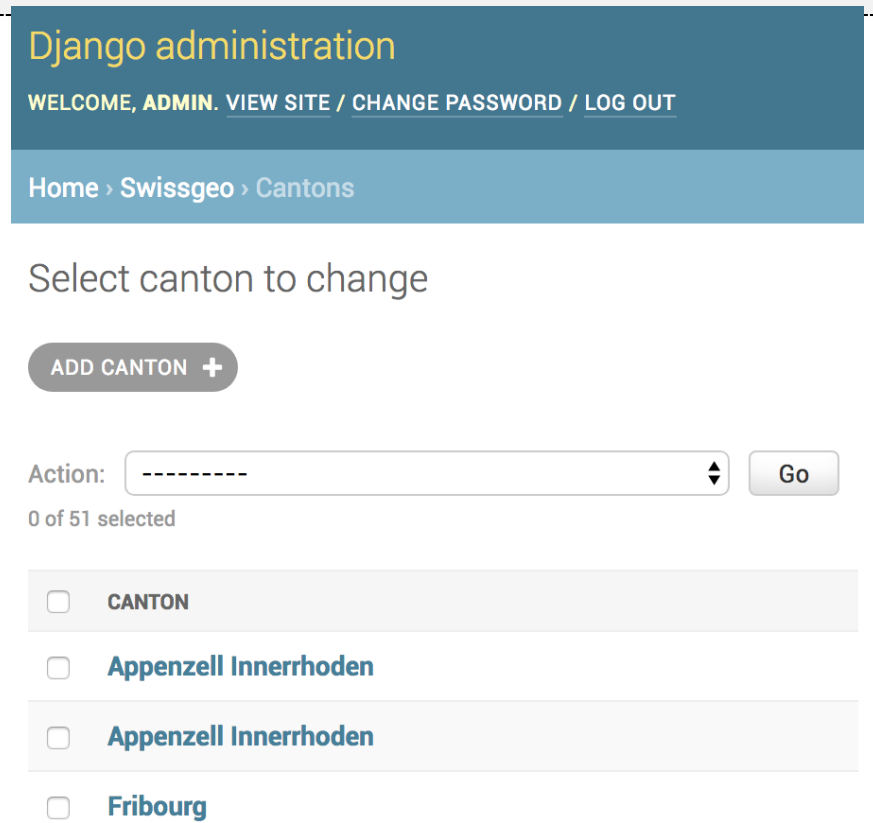


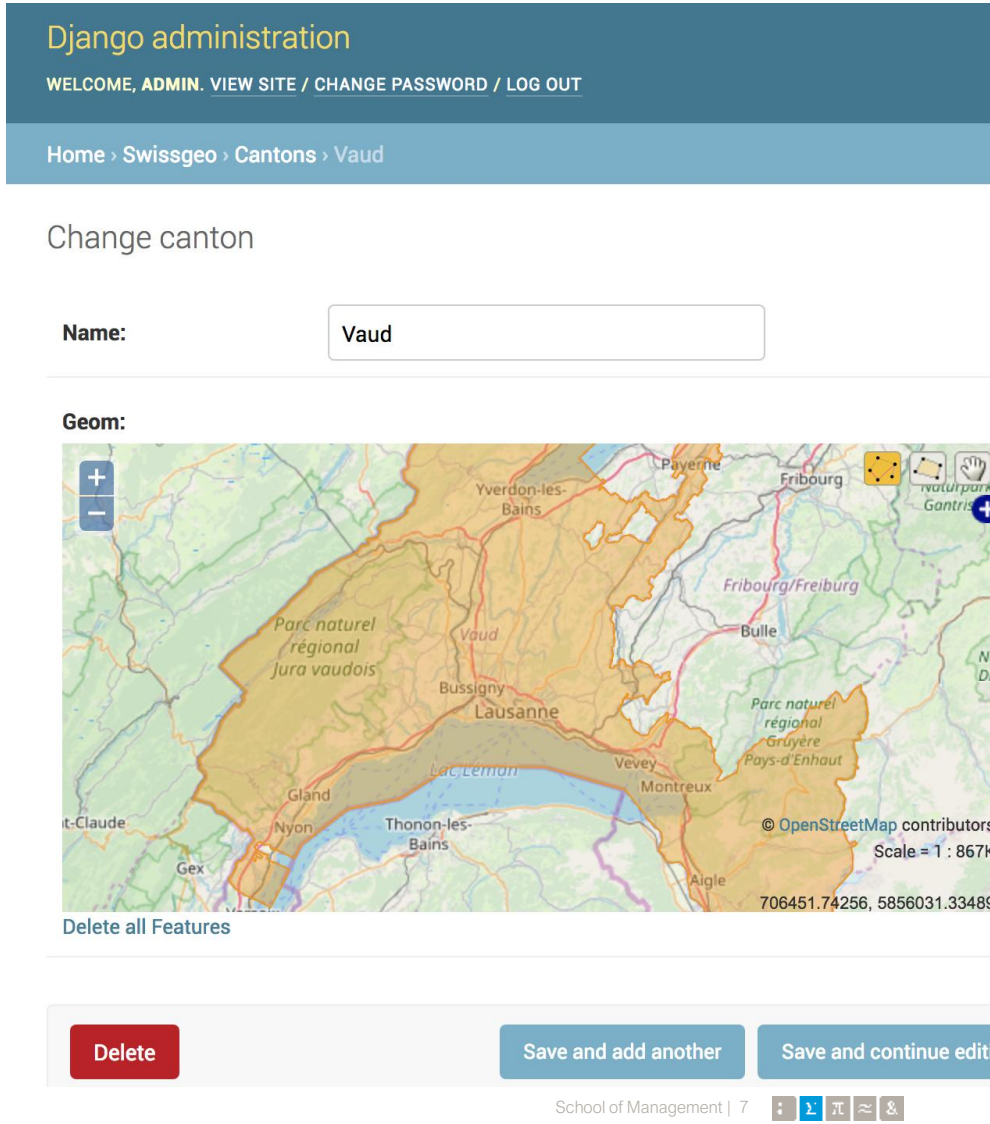


> admin.py

```
from django.contrib.gis import admin

from .models import City, Hospital, Canton
admin.site.register(City)
admin.site.register(Hospital)
admin.site.register(Canton, admin.ModelAdmin)
```





> Requests with Geo Models

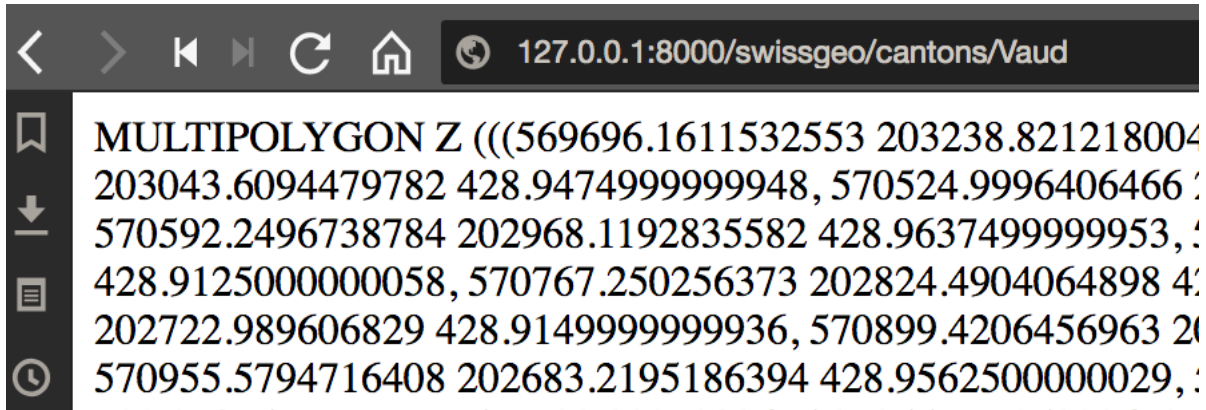
http://127.0.0.1:8000/swissgeo/cantons/Vaud

urls.py

```
urlpatterns = [
  path('city/', views.cities, name='cities'),
  path('city/<int:city_id>', views.city, name='city'),
  path('cantons/<str:canton_name>', views.canton, name='canton'),
```

views.py

```
def canton(request,canton_name):
    cantons=Canton.objects.filter(name=canton_name)
    return HttpResponse(cantons[0].geom.wkt)
```



> Templates with Geo Models

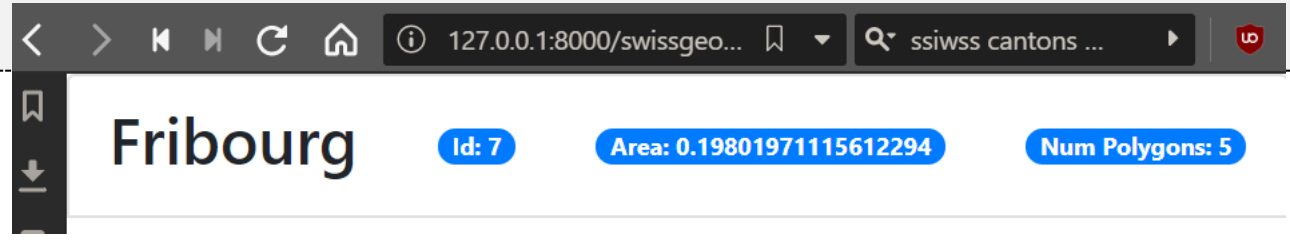
views.py

http://127.0.0.1:8000/swissgeo/cantons/Fribourg

```
def canton(request,canton_name):
  cantons=Canton.objects.filter(name=canton_name)
  return render(request,'swissgeo/canton.html',
                {'cantonobj':cantons[0]})
```

swissgeo/canton.html

```
<ul class="list-group">
  <li class="list-group-item d-flex justify-content-between align-items-center">
     <h2>{{cantonobj.name}}</h2>
     <span class="badge badge-primary badge-pill">Id:{{cantonobj.id}}</span>
     <span class="badge badge-primary badge-pill">Area: {{cantonobj.geom.area }}</span>
     <span class="badge badge-primary badge-pill"># Polygons: {{cantonobj.geom.num_geom}}</sp
  </li>
</ul>
```



> Returning GeoJSON

http://127.0.0.1:8000/swissgeo/cantons.json

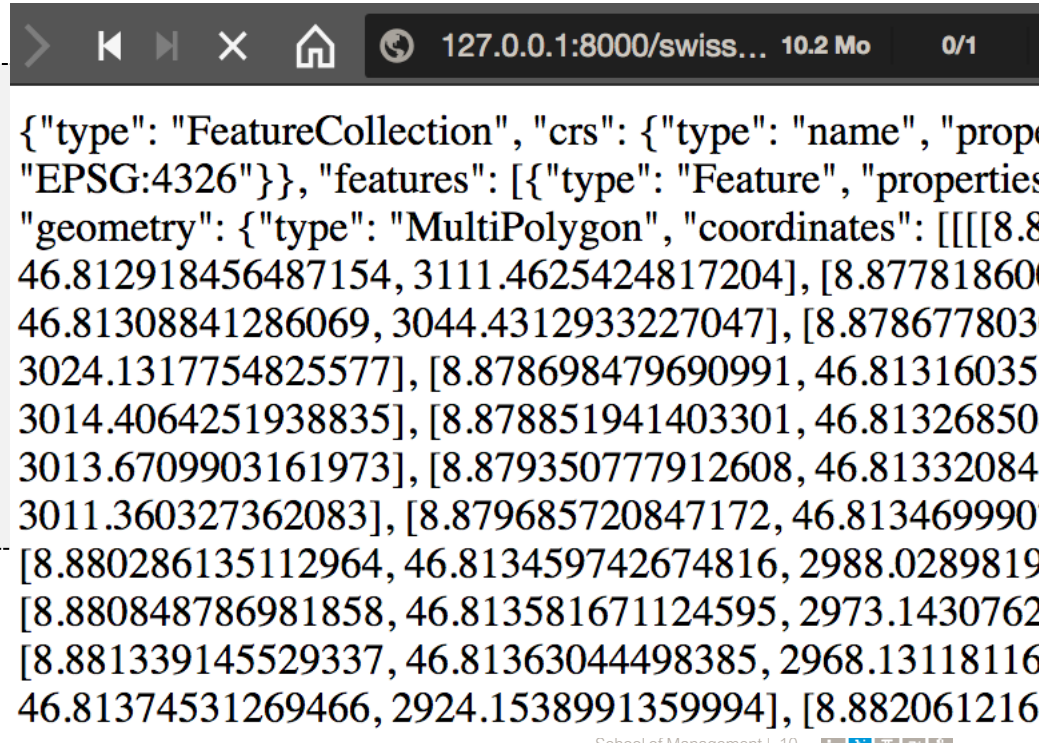
urls.py

```
urlpatterns = [
  path('cantons/<str:canton_name>', views.canton, name='canton'),
  path('cantons.json', views.cantonsjson, name='cantonsjson'),
```

views.py

```
from django.core.serializers import serialize

def cantonsjson(request):
   cantons=Canton.objects.all()
   ser=serialize('geojson',cantons,
                 geometry_field='geom',
                 fields=('name',))
   return HttpResponse(ser)
```



> Leaflet in GeoDjango

https://django-leaflet.readthedocs.io

settings.py

```
INSTALLED_APPS = [
  'swissgeo.apps.SwissgeoConfig',
  'django.contrib.admin',
  'django.contrib.auth',
  'django.contrib.contenttypes',
  'django.contrib.sessions',
  'django.contrib.messages',
  'django.contrib.staticfiles',
  'django.contrib.gis',
  'leaflet',
]
```

```
LEAFLET_CONFIG ={
   'MIN_ZOOM':3,
   'MAX_ZOOM':15,
   'DEFAULT_ZOOM':7,
   'DEFAULT_CENTER': (46.7,8),
   'TILES': [('OSM',
'//tile.openstreetmap.org/{z}/{x}/{y}.pn
g', {'attribution': '© <a
href="https://www.openstreetmap.org/copy
right">OpenStreetMap</a>
contributors','referrerPolicy': 'strict-
origin'})]

}
```

> Leaflet in django templates

urls.py

```
 urlpatterns = [
    path('cantons', views.cantons, name='cantons'),
```

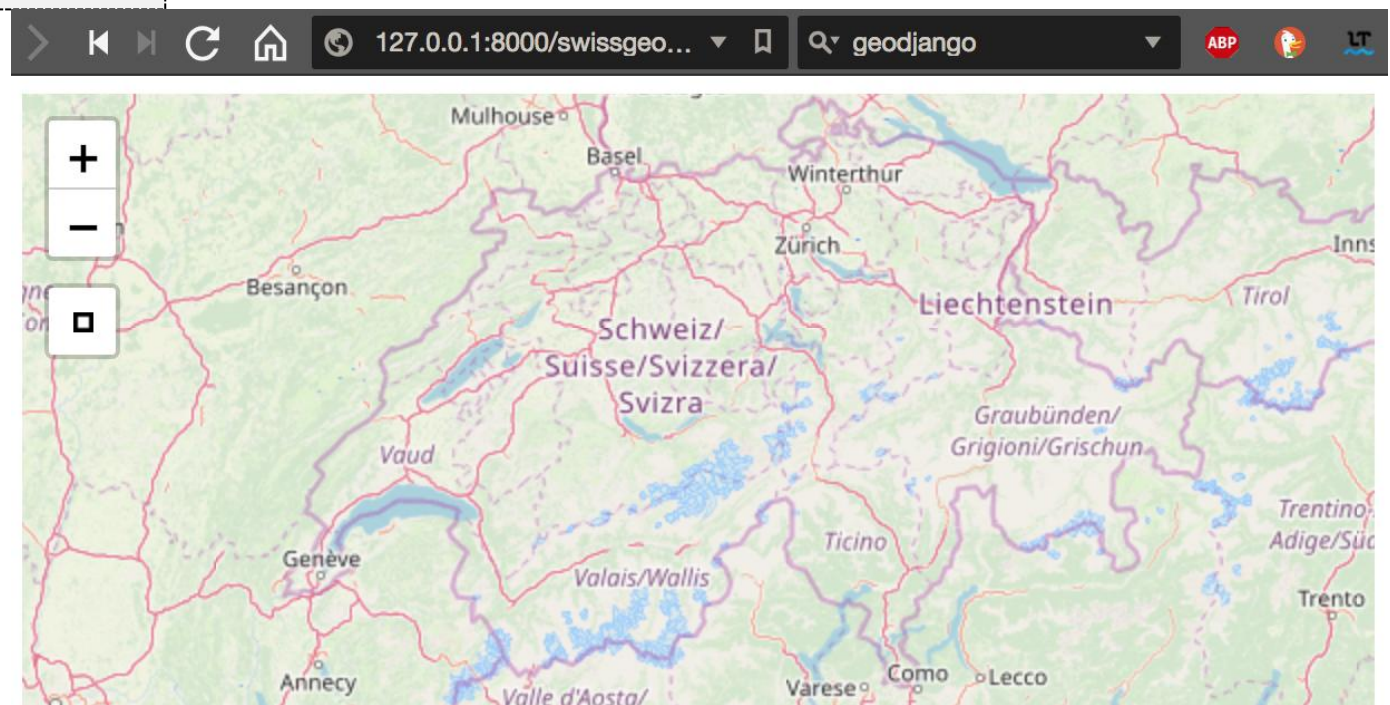
views.py

```
def cantons(request):
   context ={     }
   return render(request,
     'swissgeo/cantons.html',context)
```

cantons.html

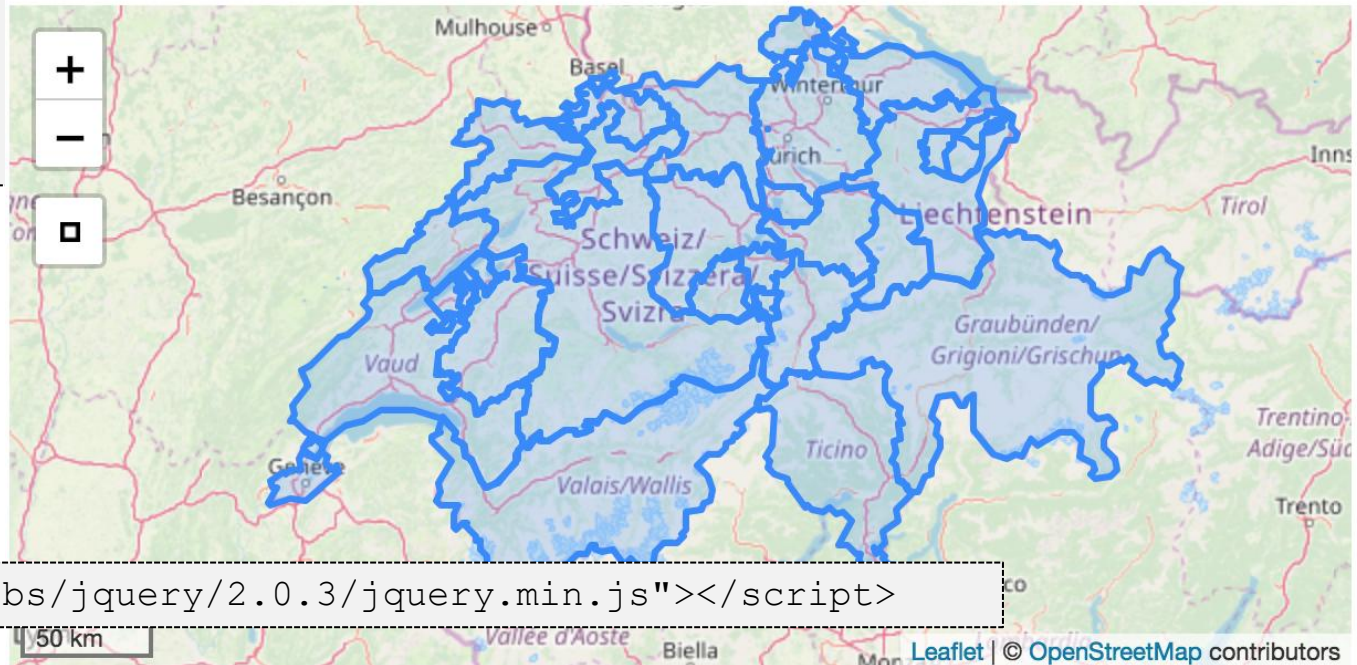
```
{% load leaflet_tags %}
<head>
{% leaflet_js %}
{% leaflet_css %}
</head>

<body>
{% leaflet_map "cantons" %}
</body>
```



> Leaflet in django templates

```
<body>
    {% leaflet_map "cantons" callback="main_map_init" %}
    <script type="text/javascript">
      function main_map_init (map, options) {
        var dataurl = '{% url "cantonsjson" %}'
        $.getJSON(dataurl, function (data){
           L.geoJson(data).addTo(map); });
        }
</script>
</body>
```

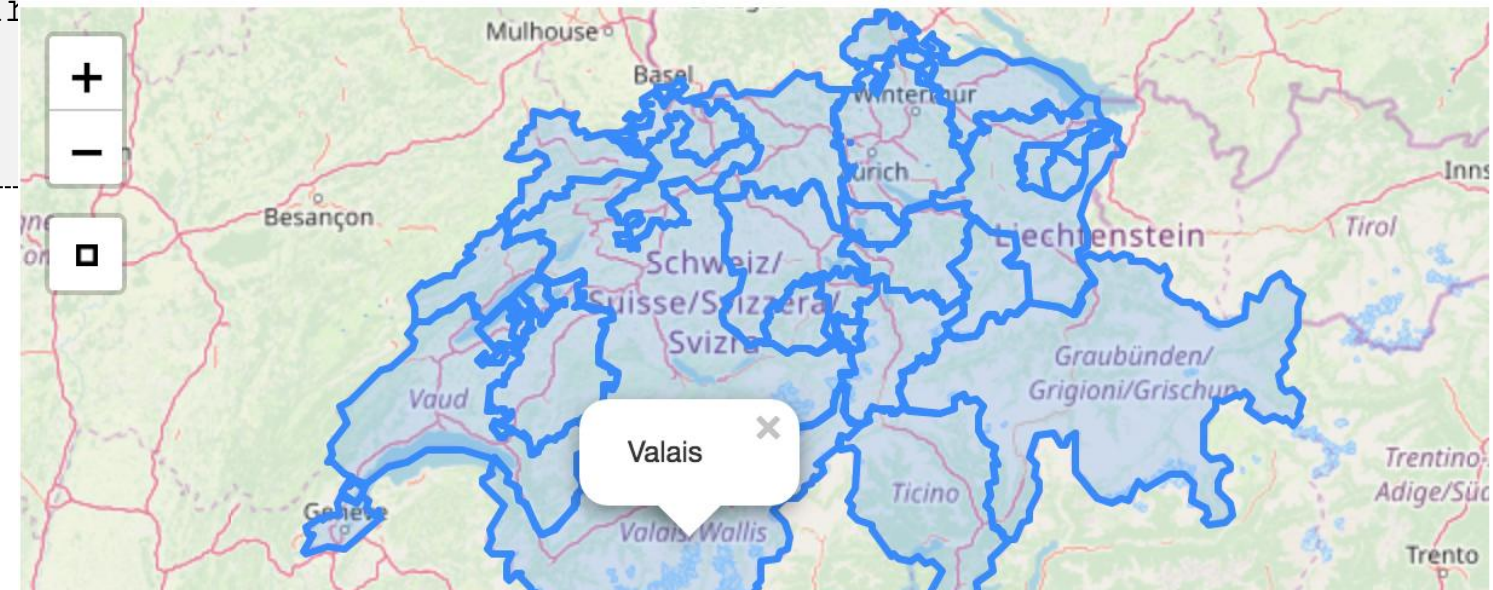


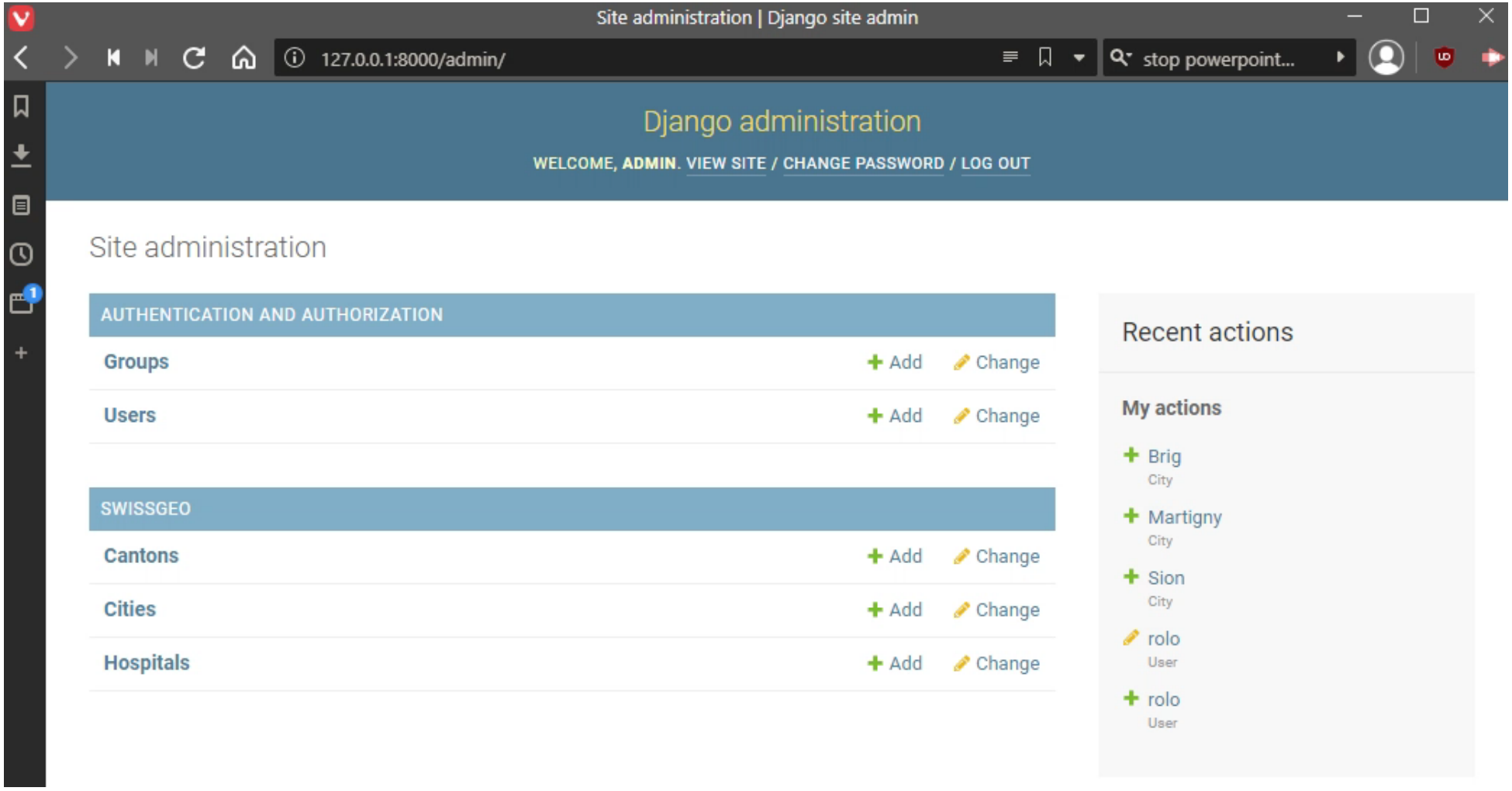

```
<script src="//cdnjs.cloudflare.com/ajax/libs/jquery/2.0.3/jquery.min.js"></script>
```

> Leaflet in django templates

```
function main_map_init (map, options) {
     var dataurl = '{% url "cantonsjson" %}'
     $.getJSON(dataurl, function (data){
          L.geoJson(data,{onEachFeature: onEachFeature}).addTo(map); });
}

function onEachFeature(feature, layer) {
    if (feature.properties &&
          feature.properties.name) {
       layer.bindPopup(featur
    }
}
```




Site administration | Django site admin
127.0.0.1:8000/admin/
stop powerpoint...
Django administration
WELCOME, ADMIN. VIEW SITE / CHANGE PASSWORD / LOG OUT
Site administration
AUTHENTICATION AND AUTHORIZATION
Groups + Add Change
Users + Add Change
SWISSGEO
Cantons + Add Change
Cities + Add Change
Hospitals + Add Change
Recent actions
My actions
+ Brig
City
+ Martigny
City
+ Sion
City
rolo
User
+ rolo
User

Thanks!
Questions?