

Conteúdo

Tema.....	1
1. Open Layers.....	1
1.1 Estrutura Base	1
1.2 OpenLayers e GeoJSON	2
1.3 OpenLayers e XML.....	3
1.4 OpenLayers e WMS as a Service	3
1.5 OpenLayers + PosGreSQL + Geoserver.....	4
2. Leaflet	5
2.1 Estrutura Base	5
2.2 Leaflet e GeoJSON	8
2.3. Leaflet with XML.....	10
2.4 Leaflet e WMS as a service.....	10
2.5 PostGIS + GeoServer + Leaflet.....	11

Tema

Exploração de visualizadores de informação georreferenciada usando Open Layers e Leaflet

1. Open Layers

1.1 Estrutura Base

index.html

```

<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>OpenLayers</title>
    <style>
      html, body, #map-container {
        margin: 0;
        height: 100%;
        width: 100%;
        font-family: sans-serif;
      }
    </style>
  </head>
  <body>
    <div id="map-container"></div>
  </body>
</html>

```

main.js

```

import 'ol/ol.css';
import {Map, View} from 'ol';
import TileLayer from 'ol/layer/Tile';
import XYZSource from 'ol/source/XYZ';
import {fromLonLat} from 'ol/proj';

new Map({
  target: 'map-container',
  layers: [
    new TileLayer({
      source: new XYZSource({
        url: 'http://tile.stamen.com/terrain/{z}/{x}/{y}.jpg'
      })
    })
  ],
  view: new View({
    center: fromLonLat([0, 0]),
    zoom: 2
  })
});

```

+

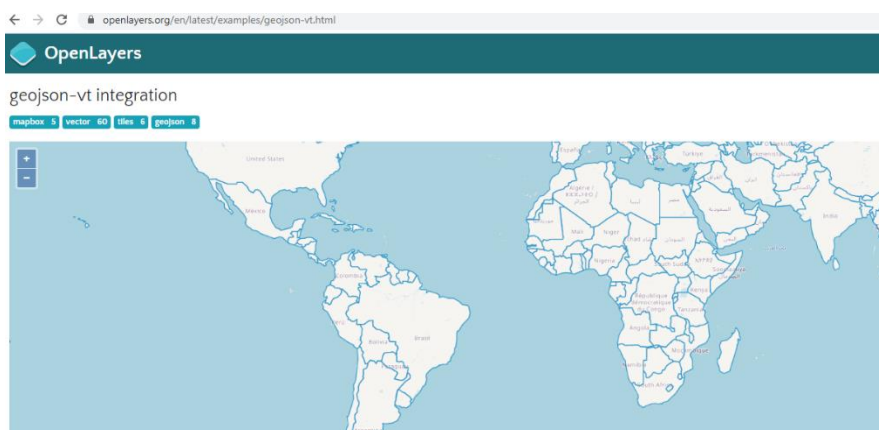
Resultado:



1.2 OpenLayers e GeoJSON

Considere o exemplo do tutorial disponível em:

- Países: <https://openlayers.org/en/latest/examples/geojson-vt.html>



- Workshop: <https://openlayers.org/workshop/en/vector/geojson.html>

Tutorial

Visualização de Informação Geográfica – Open Layers e Leaflet

(versão 1)

← → ↻ openlayers.org/workshop/en/vector/geojson.html


OpenLayers Workshop

- Introduction
- Basics
- Vector Data
 - Rendering GeoJSON
 - Drag and drop
 - Modifying features
 - Drawing new features
 - Snapping
 - Downloading features
 - Making it look nice
- Mobile Maps and Sensors
 - A mobile map
 - Geolocation
 - Compass
- WebGL Rendering
 - Map setup
 - Rendering points
 - Animating points
- Vector Tiles and Mapbox Styles
 - The VectorTile layer

```

    })
  },
  view: new View({
    center: [0, 0],
    zoom: 2
  })
});
  
```

You should now be able to see a map with country borders at <http://localhost:3000/>.



GeoJSON features

Since we'll be reloading the page a lot, it would be nice if the map stayed where we left it in a reload. We can bring in the `ol-hashed` package to make this work. This package is already installed as part of the workshop dependencies. If it were not already included, you could install it with `npm install ol-hashed`.

1.3 OpenLayers e XML


Considere o exemplo em: <https://openlayers.org/en/latest/examples/vector-osm.html>

← → ↻ openlayers.org/en/latest/examples/vector-osm.html

OpenLayers

OSM XML

vector 60 osmxml 1 loading 6 server 4 strategy 1 bbox 2 maptiler 24



OSM XML vector data is loaded dynamically from a the [Overpass API](#) using a `bbox` strategy. Note that panning and zooming will eventually lead to "Too many requests" errors from

main.js

```

1 import 'ol/ol.css';
2 import Map from 'ol/Map';
3 import OSMXML from 'ol/format/OSMXML';
4 import VectorSource from 'ol/source/Vector';
  
```

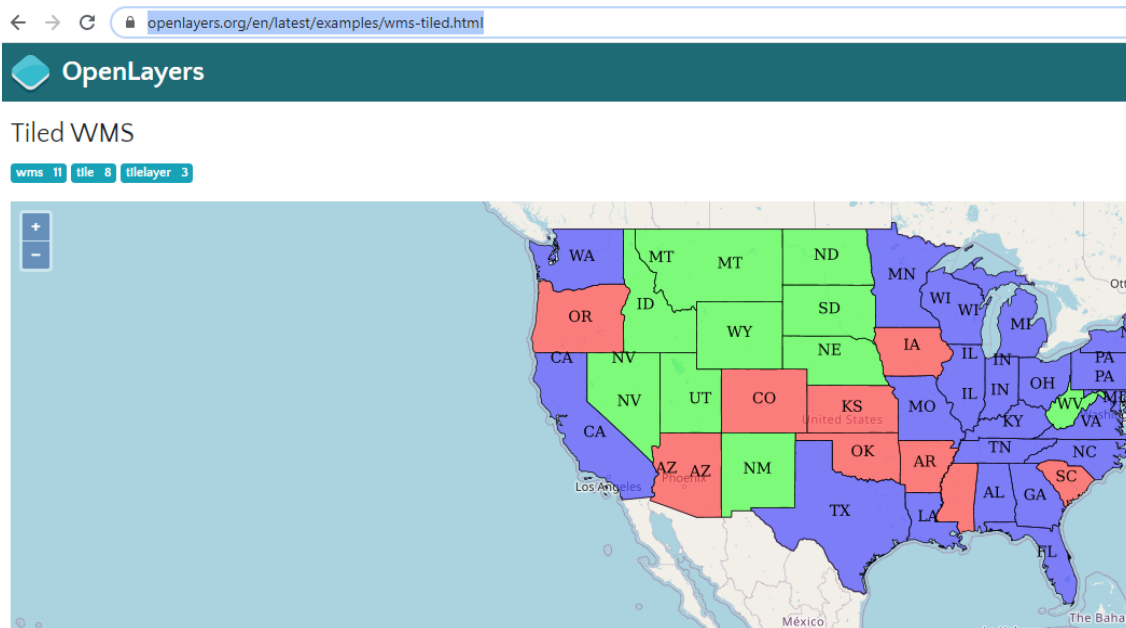
1.4 OpenLayers e WMS as a Service

Considere o exemplo em: <https://openlayers.org/en/latest/examples/wms-tiled.html>

Tutorial

Visualização de Informação Geográfica – Open Layers e Leaflet

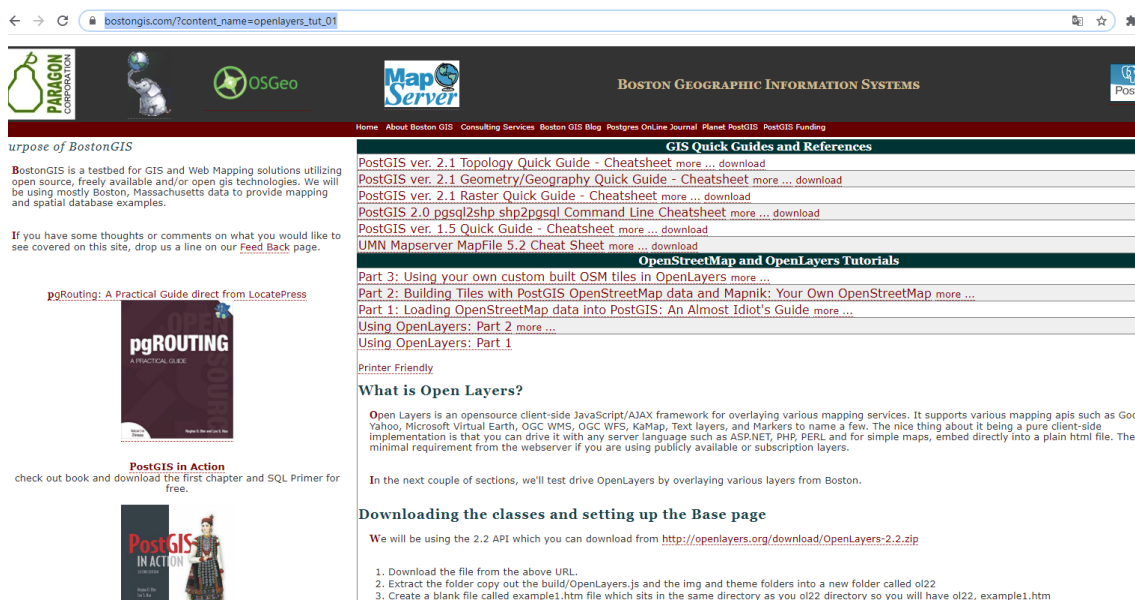
(versão 1)



1.5 OpenLayers + PosgreSQL + Geoserver

Considere os exemplos disponíveis em:

https://www.bostongis.com/?content_name=openlayers_tut_01



2. Leaflet

2.1 Estrutura Base

Considere a informação disponível no link: <https://leafletjs.com/examples.html> Seguindo estas bases, de seguida apresentam-se um conjunto de exemplos de exploração das funcionalidades do Leaflet.

Scripts para aplicar a biblioteca Leaflet num ficheiro JavaScript:

```
<link rel="stylesheet" href="https://unpkg.com/leaflet@1.3.4/dist/leaflet.css"
integrity="sha512-puBpdR07980ZvTTbP4A8Ix/l+A4dHDD0DGqYW6RQ+9jxkRfclaxxQb/SJAWZfWAKuyeQUyT07+7N4QKrDh+drA=="
crossorigin="" />

<script src="https://unpkg.com/leaflet@1.3.4/dist/leaflet.js"
integrity="sha512-nMmmRyTVoLYqjP9hrbed9S+FzjZHW5gY1TWCHA5ckwXZBadntCNS8kEqAWdrb907rxbCaA41KTIWjDXZxf10cA=="
crossorigin=""></script>
```

Estilização do mapa para ser apresentado no browser:

```
<style>

#mapid {
  height: 480px;
}

</style>
```

Declaração do mapa e criação do layer (camada que apresenta o mapa):

```
var mymap = L.map('mapid').setView([51.505, -0.09], 13);

var token = "pk.eyJ1Ijoiam51bm9mZXJyZWlyYSIsImEiOiIjam5zMGdsb3owYjFqM2txcTA2bmN0OHZwIn0.U1hgHg316EPnNvALad0oqQ";

L.tileLayer('https://api.tiles.mapbox.com/v4/{id}/{z}/{x}/{y}.png?access_token={accessToken}', {
  attribution: 'Map data &copy; <a href="https://www.openstreetmap.org/">OpenStreetMap</a> contributors, <a href="https://c',
  maxZoom: 18,
  id: 'mapbox.streets',
  accessToken: token
}).addTo(mymap);
```

Criação de um ponto, círculo e polígono no mapa:

```
var marker = L.marker([51.5, -0.09]).addTo(mymap);

var circle = L.circle([51.508, -0.11], {
  color: 'red',
  fillColor: '#f03',
  fillOpacity: 0.5,
  radius: 500
}).addTo(mymap);

var polygon = L.polygon([
  [51.509, -0.08],
  [51.503, -0.06],
  [51.51, -0.047]
]).addTo(mymap);
```

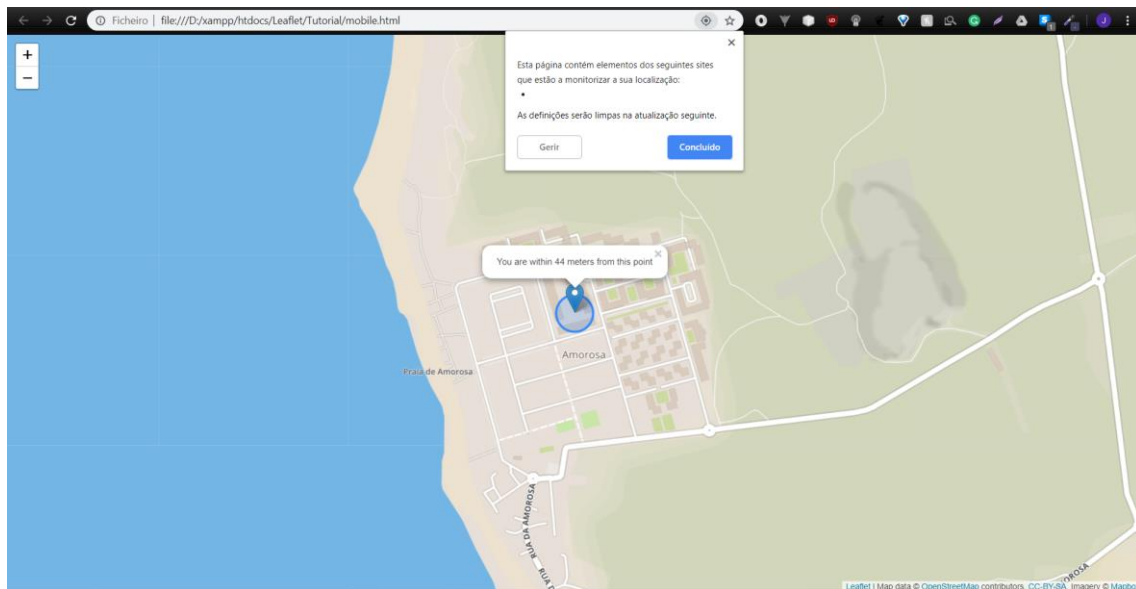
Atribuição de popups aos elementos criados:

```
marker.bindPopup("<b>Hello world!</b><br> I am a popup.");
circle.bindPopup("I am a circle");
polygon.bindPopup("I am a polygon");
```

Função de clique que permite apresentar a latitude e longitude do ponto:

```
function onMapClick(e){
  popup
    .setLatLng(e.latlng)
    .setContent("You clicked the map at " + e.latlng.toString())
    .openOn(mymap);
}
```

Resultado:



Criação dos layers (light, streets and satellite):

```
//Definimos 3 Layers default (Grayscale, Streets e Satellite)
var grayscale = L.tileLayer('https://api.tiles.mapbox.com/v4/{id}/{z}/{x}/{y}.png?access_token={accessToken}', {id: 'mapbox.light', accessToken: 'pk.eyJ1IjoiaWoi...
streets = L.tileLayer('https://api.tiles.mapbox.com/v4/{id}/{z}/{x}/{y}.png?access_token={accessToken}', {id: 'mapbox.streets', accessToken: 'pk.eyJ1IjoiaWoi...
satellite = L.tileLayer('https://api.tiles.mapbox.com/v4/{id}/{z}/{x}/{y}.png?access_token={accessToken}', {id: 'mapbox.satellite', accessToken: 'pk.eyJ1IjoiaWoi...

```

Adicionamos a atributos pré-definidos do Leaflet as 3 layers criadas:

Tutorial

```
//Dizemos que as 3 Layers default estão dentro da variavel "baseMaps"
var baseMaps = {
  "Grayscale": grayscale,
  "Streets": streets,
  "Satellite": satellite
};
```

Criação do mapa e atribuímos as layers que serão apresentadas na primeira instância:

```
//Pomos o Mapa (já preparado com as layers acima definidas) associado ao div com o id "mapid"
var mymap = L.map('mapid', {
  center: [41.50, -7.73],
  zoom: 9,
  layers: [streets, carros] //aqui definimos as layers que queremos que estejam visiveis numa primeira instancia
});
```

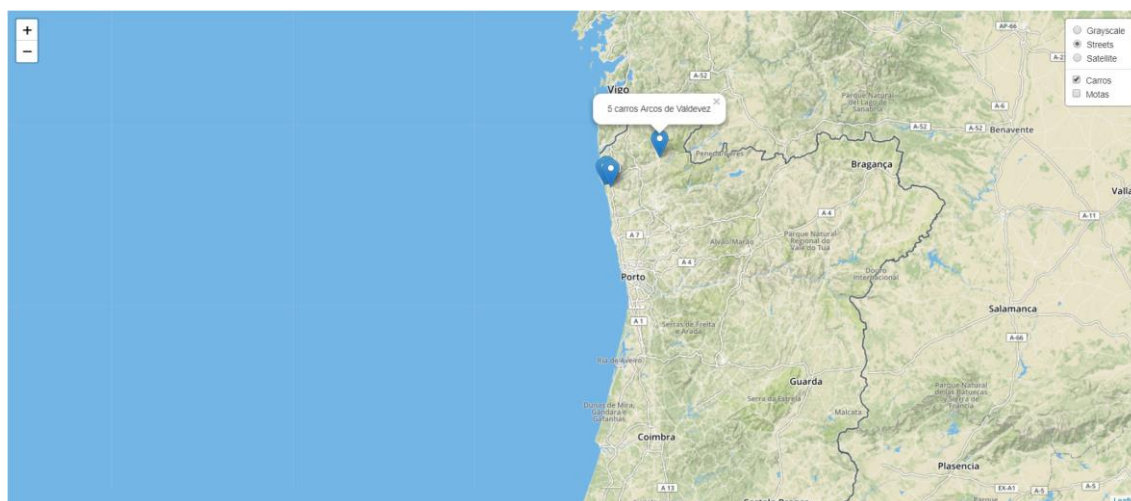
Criação das layers que podem ser apresentadas em simultâneo (overlay):

```
//Aqui definimos as restantes layers (para além das default). Podemos mostrar várias ao mesmo tempo
var overlayMaps = {
  "Carros": carros,
  "Motas": motas
};
```

Adição de um controlador na página para se visualizar as layers criadas:

```
//No canto superior direito aparece um ocntrolador para por as layers visiveis ou invisiveis
L.control.layers(baseMaps, overlayMaps).addTo(mymap);
```

Resultado:

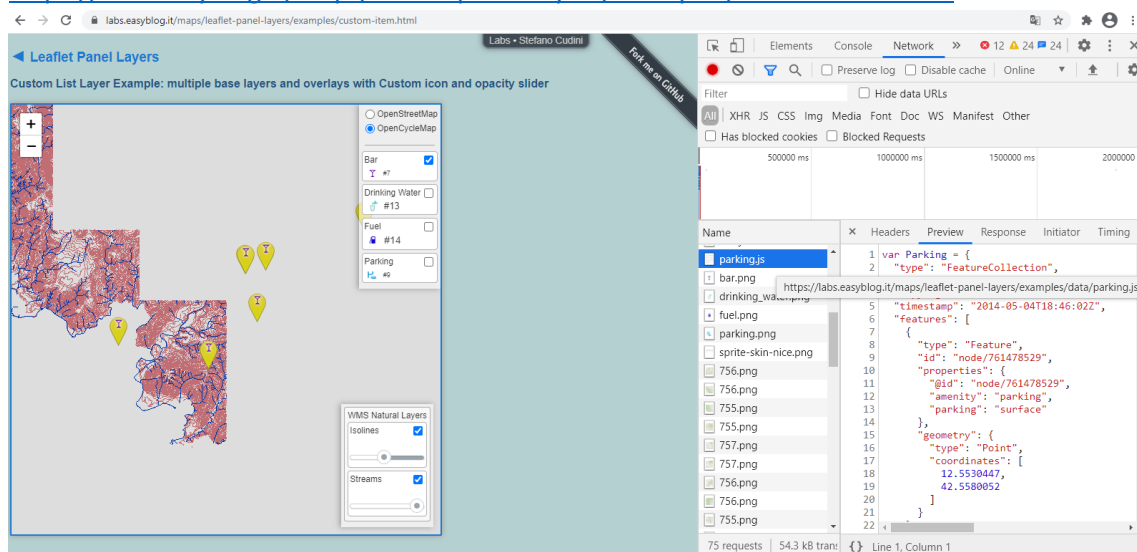


Considere os seguintes exemplos, seguindo os tutoriais disponibilizados nos seguintes links:

Tutorial

Visualização de Informação Geográfica – Open Layers e Leaflet

(versão 1)

<https://labs.easyblog.it/maps/leaflet-panel-layers/examples/custom-item.html>

2.2 Leaflet e GeoJSON

Considere os exemplos em:

<https://leafletjs.com/examples/geojson/>

leafletjs.com/examples/geojson/



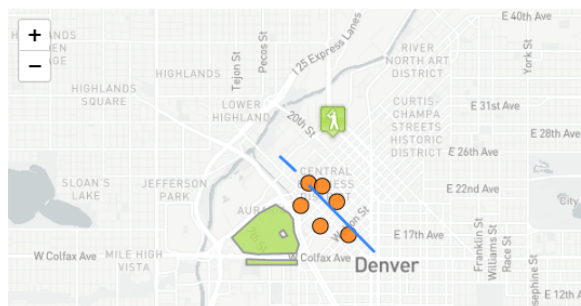
an open-source JavaScript library
for mobile-friendly interactive maps

[Overview](#) [Tutorials](#) [Docs](#) [Download](#) [Plugins](#) [Blog](#)

[Tutorials](#)

Using GeoJSON with Leaflet

GeoJSON is becoming a very popular data format among many GIS technologies and services — it's simple, lightweight, straightforward, and Leaflet is quite good at handling it. In this example, you'll learn how to interact with map vectors created from [GeoJSON](#) objects.



Poderá executar também o seguinte tutorial: <http://duspviz.mit.edu/web-map-workshop/leaflet-javascript-interactions/>

← → 🔒 Não seguro | duspviz.mit.edu/web-map-workshop/leaflet-javascript-interactions/

HOME CALENDAR TUTORIALS RESOURCES CONTACT

Click save and continue to check out our new toggle button and witness the power of an `if... else` statement.

Our Map

Toggle Coffee Shops (Click to view this example on its own.)

2.3. Leaflet with XML

Maps from a Database: Reading XML data into Leaflet: http://erica.altschul.info/Tutorial_XML-to-Leaflet.pdf

2.4 Leaflet e WMS as a service

Considere o exemplo do link: <https://leafletjs.com/examples/wms/wms.html>

leafletjs.com/examples/wms/wms.html



[See this example stand-alone.](#)

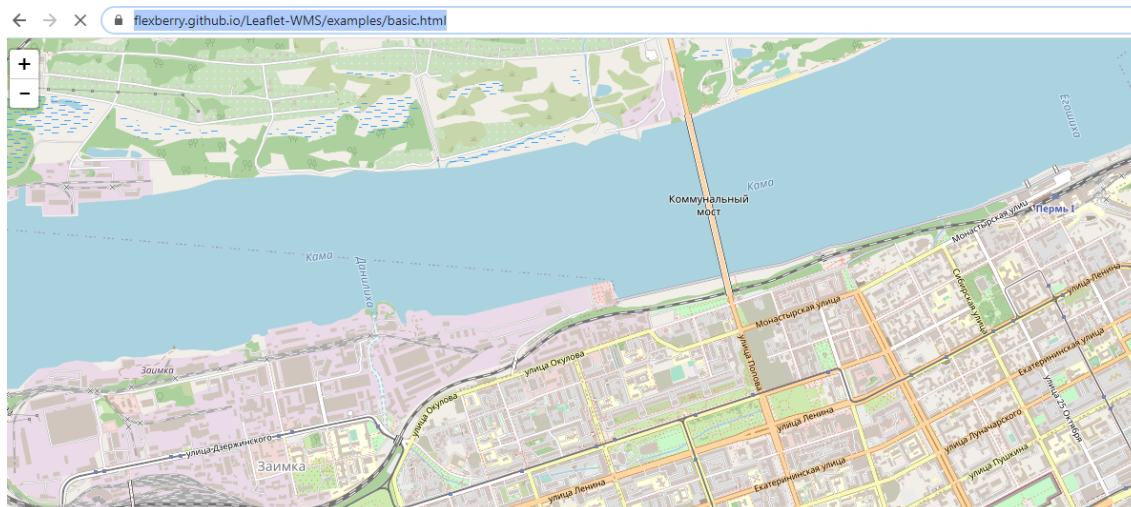
Or we can try the SRTM30-Colored-Hillshade WMS layer:

```
var wmsLayer = L.tileLayer.wms('http://ows.mundialis.de/services/service?', {  
  layers: 'SRTM30-Colored-Hillshade'  
}).addTo(map);
```

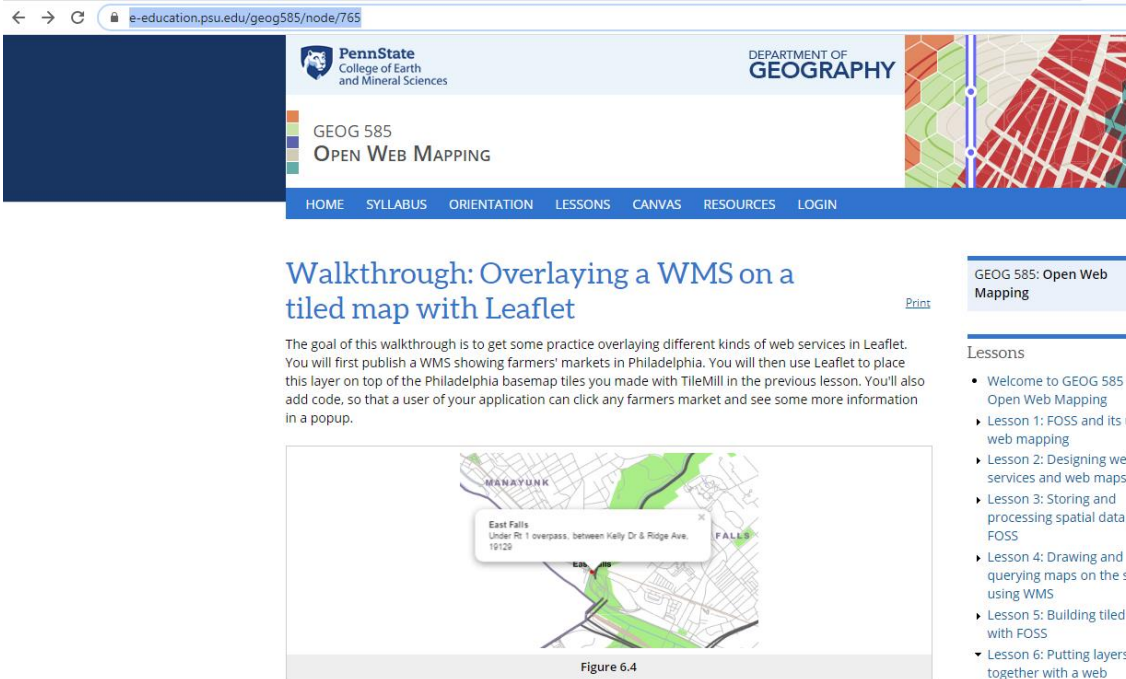


[See this example stand-alone.](#)

Considere também o seguinte exemplo: <https://flexberry.github.io/Leaflet-WMS/examples/basic.html>



Considere também o exemplo seguinte: <https://www.e-education.psu.edu/geog585/node/765>



Walkthrough: Overlaying a WMS on a tiled map with Leaflet

The goal of this walkthrough is to get some practice overlaying different kinds of web services in Leaflet. You will first publish a WMS showing farmers' markets in Philadelphia. You will then use Leaflet to place this layer on top of the Philadelphia basemap tiles you made with TileMill in the previous lesson. You'll also add code, so that a user of your application can click any farmers market and see some more information in a popup.

Figure 6.4

Setting up the farmers' markets WMS

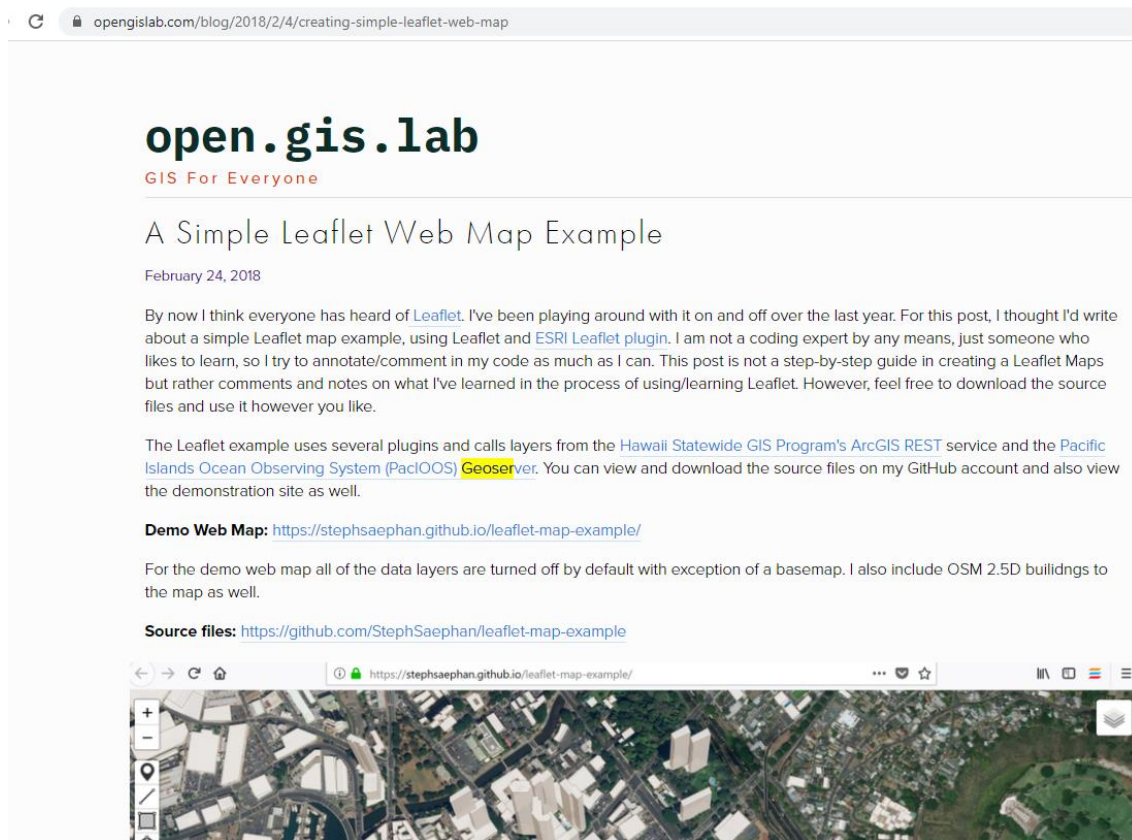
LESSONS

- Welcome to GEOG 585 - Open Web Mapping
- ▶ Lesson 1: FOSS and its use in web mapping
- ▶ Lesson 2: Designing web services and web maps
- ▶ Lesson 3: Storing and processing spatial data with FOSS
- ▶ Lesson 4: Drawing and querying maps on the server using WMS
- ▶ Lesson 5: Building tiled maps with FOSS
- ▼ Lesson 6: Putting layers together with a web mapping API
 - Overview

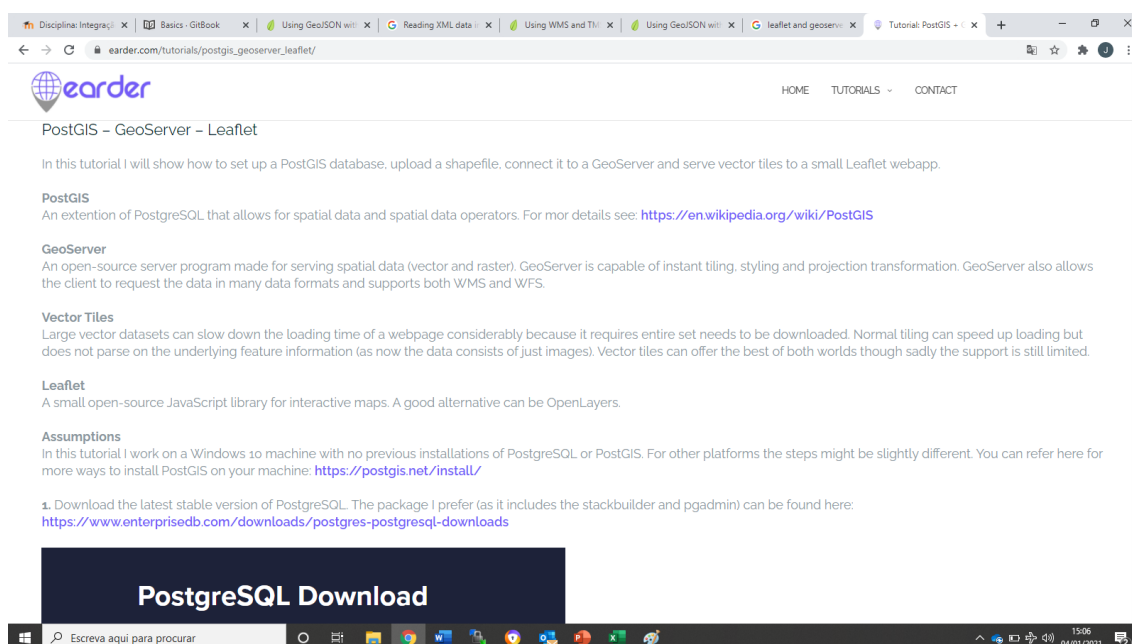
2.5 PostGIS + GeoServer + Leaflet

Considere o exemplo de exploração do Leaflet com Geoserver, disponível em:

<https://opengislab.com/blog/2018/2/4/creating-simple-leaflet-web-map>



Considere o exemplo de exploração do Leaflet com Geoserver e PostgreSQL, disponível em:
https://www.earder.com/tutorials/postgis_geoserver_leaflet/



Tutorial

Visualização de Informação Geográfica – Open Layers e Leaflet

(versão 1)

<https://labs.easyblog.it/maps/gpx-simplify-optimizer/>