

4. Plugins Leaflet



Leaflet Plugins

- Plugins <http://leafletjs.com/plugins.html>

Plugin Leaflet-search: Ejemplo buscador de Farmacias

Plugin que permite buscar dentro los atributos de un GeoJson

- Creamos archivo **farmacias.html**
- Visualizamos geojson en [/geoweb/datos/farmacias.geojson](#)
- Utilizaremos directamente los plugins: **GeoJSON AJAX** <https://calvinmetcalf.github.io/leaflet-ajax/dist/leaflet.ajax.js> Leaflet-Search <https://github.com/stefanocudini/leaflet-search>

```

<html lang="es">

<head>
  <title>Farmacias</title>
  <meta charset="utf-8" />
  <meta name="viewport" content="width=device-width, initial-
scale=1.0">
  <meta name="author" content="autor" />
  <meta name="description" content="descripción página">
  <meta name="robots" content="index,follow">
  <link rel="stylesheet" href="https://unpkg.com/leaflet@1.4.0/
dist/leaflet.css" />
  <script>
    L_PREFER_CANVAS = true;
  </script>
  <script src="https://unpkg.com/leaflet@1.4.0/dist/
leaflet.js"></script>
  <script src="https://calvinmetcalf.github.io/leaflet-ajax/
dist/leaflet.ajax.js"></script>
  <!-- Paso 2
    <script src="https://labs.easyblog.it/maps/leaflet-search/src/
leaflet-search.js"></script>
    <link rel="stylesheet" href="https://labs.easyblog.it/maps/
leaflet-search/src/leaflet-search.css" />
  -->

  <style>
    body {
      margin: 0;
      padding: 0;
      overflow: hidden;
    }

    #map {
      height: 100%;
      width: 100%;
    }
  </style>
  <script>
    var map, osm, esri;
    var geojson, farmacias;
    var controlCapas;
    var controlEscala;

    function Init() {

      map = L.map('map', {

```

```

        center: [41.3954, 2.16859],
        zoom: 14
    });
    esri = L.tileLayer(
        'http://server.arcgisonline.com/ArcGIS/rest/
services/World_Imagery/MapServer/tile/{z}/{y}/{x}', {
        maxZoom: 17,
        minZoom: 1,
        attribution: 'Tiles © Esri',
    }).addTo(map);

    osm = L.tileLayer('http://{s}.tile.openstreetmap.org/
{z}/{x}/{y}.png', {
        maxZoom: 19,
        minZoom: 1,
        attribution: 'OSM'
    });

/* Pas1
    farmacias = new L.GeoJSON.AJAX('datos/
farmacias.geojson', {
        maxZoom: 19,
        minZoom: 14,
        onEachFeature: function (feature, layer) {
            popupContent = "<b>" + feature.properties.NOM
+ "</b><br>" + feature.properties
                .CARRCADAST + " " +
feature.properties.DOORNUM + "</b>";
            layer.bindPopup(popupContent);
        },
        pointToLayer: function (feature, latlng) {
            return L.circleMarker(latlng, {
                radius: 6,
                fillColor: "#00ff00",
                color: "#ffffff",
                weight: 3,
                opacity: 1,
                fillOpacity: 0.8
            });
        }
    }).addTo(map);
*/

var baseMaps = {
    "Orto_esri": esri,
    "Mapa_osm": osm
};

/*Pas 1
    var overlayMaps = {
        "farmacias": farmacias

```

```

    };
*/
    controlCapas = L.control.layers(baseMaps, null);
    controlCapas.addTo(map);
    controlEscala = L.control.scale();
    controlEscala.addTo(map);

/*Pas 2
    var searchControl = new L.Control.Search({
        layer: farmacias,
        propertyName: 'NOM',
        circleLocation: true,
        moveToLocation: function (latlng) {

            map.setView(latlng, 17);
        }
    });

    map.addControl(searchControl);
*/

    };
</script>
</head>

<body onload="Init()">
    <div id="map"></div>
</body>

</html>

```

Plugin Geosearch: Ejemplo buscador de Callejero

Plugin que permite conectar con servicios de Geocodificación * Creamos archivo **leaflet-calles.html** * Descargaremos plugin <https://github.com/MuellerMatthew/L.GeoSearch>

```
<html lang="es">

<head>
  <title>Calles</title>
  <meta charset="utf-8" />
  <meta name="viewport" content="width=device-width, initial-
scale=1.0">
  <meta name="author" content="autor" />
  <meta name="description" content="descripción página">
  <meta name="robots" content="index,follow">
  <link rel="stylesheet" href="https://unpkg.com/leaflet@0.7.7/
dist/leaflet.css" />
  <link rel="stylesheet" href="https://unpkg.com/leaflet-
geosearch@2.7.0/assets/css/leaflet.css" />

  <script>
    L_PREFER_CANVAS = true;
  </script>
  <script src="https://unpkg.com/leaflet@0.7.7/dist/
leaflet.js"></script>
  <script src="https://unpkg.com/leaflet-geosearch@2.7.0/dist/
bundle.min.js"></script>

  <style>
    body {
      margin: 0;
      padding: 0;
      overflow: hidden;
    }

    #map {
      height: 100%;
      width: 100%;
    }

    .leaflet-control-geosearch .results > * {

      cursor: pointer
    }
  </style>
  <script>
    var map, osm, esri;
    var geojson, farmacias;
    var controlCapas;
    var controlEscala;
```

```

function Init() {

    map = L.map('map', {
        center: [41.3954, 2.16859],
        zoom: 14
    });

    esri = L.tileLayer(
        'http://server.arcgisonline.com/ArcGIS/rest/
services/World_Imagery/MapServer/tile/{z}/{y}/{x}', {
        maxZoom: 17,
        minZoom: 1,
        attribution: 'Tiles © Esri',
    }).addTo(map);

    osm = L.tileLayer('http://{s}.tile.openstreetmap.org/
{z}/{x}/{y}.png', {
        maxZoom: 19,
        minZoom: 1,
        attribution: 'OSM'
    });

    var baseMaps = {
        "Orto_esri": esri,
        "Mapa_osm": osm
    };

    controlCapas = L.control.layers(baseMaps, null);
    controlCapas.addTo(map);

    new GeoSearch.GeoSearchControl({
        // provider: new
GeoSearch.OpenStreetMapProvider()
        provider: new GeoSearch.EsriProvider()
    }).addTo(map);

};
</script>
</head>

<body onload="Init()">
    <div id="map"></div>
</body>

```

```
</html>
```


Servicio GeoNames y Leaflet

[Geonames.org](https://www.geonames.org/) es una web que nos ofrece hasta 34 servicios geográficos Nos damos de Alta en GeoNames

- Crearemos el archivo **geonames-terremotos.html**

```

<html>

<head lang="es">
  <title>GeoNames</title>
  <meta http-equiv="Content-Type" content="text/html;
charset=UTF-8">
  <meta name="viewport" content="width=device-width, initial-
scale=1">
  <link rel="stylesheet" href="https://cdnjs.cloudflare.com/
ajax/libs/leaflet/1.4.0/leaflet.css">
  <script src="https://ajax.googleapis.com/ajax/libs/jquery/
3.2.1/jquery.min.js"></script>
  <script src="https://cdnjs.cloudflare.com/ajax/libs/leaflet/
1.4.0/leaflet.js"></script>
  <style>
    body {
      margin: 0;
      padding: 0;
      overflow: hidden;
    }

    #map {
      height: 100%;
      width: 100%;
    }

    #ventana {
      position: absolute;
      top: 100px;
      left: 10px;
      z-index: 1000;
    }
  </style>
</head>

<body>
  <div id="map"></div>
  <script>
    var terremotoPunto = null;
    var map;
    $(document).ready(function () {

      map = L.map("map", {
        attributionControl: false,
        zoom: 8,
        center: [42, 2]
      });
  </script>

```

```

        L.tileLayer('://{s}.tile.stamen.com/toner-lite/{z}/{x}/{y}.png', {
            attribution: 'Map tiles by <a href="http://stamen.com">Stamen Design</a>, <a href="http://creativecommons.org/licenses/by/3.0">CC BY 3.0</a> – Map data © <a href="http://openstreetmap.org">OpenStreetMap</a> contributors, <a href="http://creativecommons.org/licenses/by-sa/2.0/">CC-BY-SA</a>',
            subdomains: 'abcd',
            maxZoom: 17,
            minZoom: 2
        }).addTo(map);

function peticionTerremotos() {
    var peticion = 'http://api.geonames.org/earthquakesJSON?' +
        'north=' + map.getBounds()._northEast.lat +
        '&' +
        'south=' + map.getBounds()._southWest.lat +
        '&' +
        'east=' + map.getBounds()._northEast.lng + '&' +
        'west=' + map.getBounds()._southWest.lng + '&' +
        'maxRows=50&' +
        'username=masterupc&';
    $.ajax({
        url: peticion,
        method: "GET",
        dataType: "jsonp",
        success: function (respuesta) {
            respuestaTerremotos(respuesta);
        }
    }); //fin ajax
} //fin peticion

function respuestaTerremotos(respuesta) {
    if (respuesta == null) {
        return;
    } else {
        if (terremotoPunto) {
            map.eachLayer(function (layer) {
                if (layer._radius) {
                    map.removeLayer(layer);
                }
            });
        }
    }
}

```

```

        }
    });
}
var total_terremotos = respuesta.earthquakes;
for (var i = 0; i < total_terremotos.length;
i++) {
    var terremoto = total_terremotos[i];
    terremotoPunto = new
L.circleMarker([terremoto.lat, terremoto.lng], {
    radius: parseInt(terremoto.magnitude *
2),

    fillColor: "#aa0808",
    color: "#ffffff",
    weight: 2,
    opacity: 1,
    fillOpacity: 0.8
});
    terremotoPunto.bindPopup("Mg:" +
terremoto.magnitude + "<br>" + terremoto.datetime);
    terremotoPunto.addTo(map);

}
}
} //Fin respuesta terremotos

peticionTerremotos();

map.on('moveend', function () {
    peticionTerremotos();
});

}); //fin document ready
</script>
</body>

</html>

```