

# OPEN STREET MAP, LEAFLET Y EL PLUGIN MARKER CLUSTER

Arlem Gabriel Brenes Aguilar  
e-mail: jogabra16@gmail.com

**RESUMEN:** Actualmente se encuentra en auge el desarrollo de aplicaciones web, para ello se usan distintas herramientas dependiendo de las necesidades del cliente final. OpenstreetMap y leaflet son herramientas gratuitas específicamente para el diseño de mapas en HTML, que poseen una serie de funcionalidades que facilitan la manipulación de este tipo de datos, una de ellas se conoce como Marker Cluster, que es un plugin que permite manejar grupos de marcadores para realizar distintas finalidades con ellos y Routine Machine, que permite trazar rutas en el mapa dados algunos marcadores.

**PALABRAS CLAVE:** OSM, Leaflet, Marker Cluster

## 1 INTRODUCCIÓN

El presente escrito incluye historia, información general y básica para el manejo de mapas usando las herramientas de OSM y Leaflet, así como algunos ejemplos de su uso a través de la experiencia relacionada con un ejercicio utilizando métodos básicos, plugin de routine machine y plugin de Marker Cluster.

## 2 OPEN STREET MAPS

### 2.1 HISTORIA

En el 2004, Steven Coast funda OpenStreetMaps, la idea surgió dado que en Gran Bretaña la agencia cartográfica Ordnance Survey, cobraba altos precios por consultar sus datos de información geográfica. Esta organización desde el principio fue fundada sin ánimos de lucro, con la intención de fomentar el crecimiento, desarrollo y distribución de datos geospaciales libres y permitir proveer y compartir estos datos. Después de su fundación, distintas empresas y organizaciones comienzan a donar información para el crecimiento de la plataforma. El primero de ellos fue Yahoo! En el año 2006, el cual permitió que OpenStreetMaps utilizara sus fotografías aéreas como base para la producción de mapas, este servicio estuvo activo hasta el cierre del API de Yahoo Maps. Posteriormente en el año 2007 una empresa de los países bajos llamada Automotive Navigation Data dona al proyecto un conjunto completo de datos sobre la geografía de los países bajos y principales carreteras de los países de India y China.

Para ese mismo año, OSM había alcanzado 9000 usuarios registrados en la plataforma y para el mes de agosto la organización puso en marcha un proyecto independiente llamado OpenAerialMap con el objetivo de obtener una base de datos con fotografías aéreas disponibles para su distribución de manera libre. Dos

meses después, OSM completa la importación de geodatos de la oficina de censos de los Estados Unidos. Estos terminaron convirtiéndose en los primeros pequeños pasos de su crecimiento, generando curiosidad y a su vez confiabilidad por la cantidad de datos con la que contaban hasta el momento, lo que provocó que la Universidad de Oxford se interesara en el proyecto y terminó convirtiéndose en la primera institución importante en hacer uso de los datos de OSM. Posteriormente a estos datos históricos, la organización inicia su despliegue recibiendo donaciones de otras fundaciones que harían uso de sus datos como CloudMade y Wikimedia.

Además otras organizaciones como la Agencia Catastral Francesa y Bing terminaron de proporcionar más datos para el crecimiento y robustez de la plataforma., es así como en el 2012, empresas de renombre como Apple y Foursquare empezaban a migrar a esta plataforma dejando atrás a Google Maps esto debido un cambio en sus políticas pues decidieron comenzar a cobrar por el uso del servicio mientras que OSM se mantuvo de manera gratuita y a través del tiempo generó un impacto relevante y solidez en su servicio propiciando así la alianza de muchas empresas de renombre y su migración a esta nueva herramienta. [1]

### 2.2 ¿QUÉ ES OPEN STREET MAPS?

Ahora bien, Open Street Maps o OSM, conocido así por sus siglas, es uno de los servicios mas populares para el manejo de mapas en línea orientado a la creación, publicación y visualización de datos geográficos con código libre y abierto para cualquier persona que desee colaborar en el proyecto, por tanto esta organización se encuentra creada por una gran comunidad de colaboradores que contribuyen añadiendo y manteniendo los datos sobre caminos, cafeterías, estaciones de ferrocarril, estaciones de buses, restaurantes y muchas otras opciones que se encuentran en todo el mundo. Su impacto en las tecnologías es a través del impulso mediante geodatos a distintas plataformas como aplicaciones móviles, sitios web y dispositivos de hardware. Para efectos de este escrito nos centraremos en el uso de OSM en aplicaciones web, lo cual nos permitirá fusionar otras herramientas (Leaflet, Routine Machine, Marker Cluster) para realizar distintas tareas.[2]

### 2.3 EDITANDO EL MAPA

Como se ha mencionado anteriormente, OSM se mantiene debido a la colaboración de muchas personas alrededor del mundo, es por ello que la organización pone a disposición la edición del mapa mediante su aplicación web la cual puede ser accedida [aquí](#), En dicha aplicación

solamente basta registrarnos ver Figura 1, para posteriormente iniciar con el proceso de edición del mapa.

Figura 1. Registro en OSM.

Una vez registrado, podremos acceder para realizar las ediciones del mapa. Simplemente damos click en el botón de Edit (Ver Figura 2) en la parte superior izquierda.

Figura 2. Opción editar.

Esto desplegará una opción que dice point (Ver Figura 3), damos click y presionamos en el mapa el lugar donde queremos colocar el marcador.

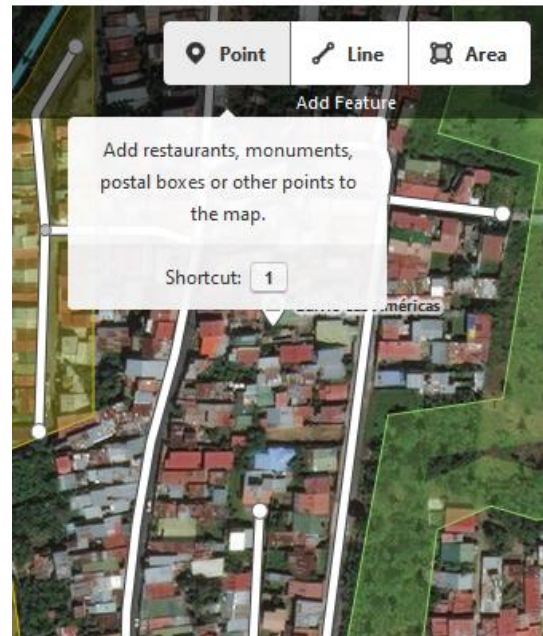


Figura 3. Opción de agregar un punto

Una vez colocado, resta ingresar la información sobre el lugar que deseamos agregar, confirmamos el cambio en el check que se encuentra al lado superior derecho de la edición de descripción del punto (Ver Figura 4).

Figura 4. Agregar descripción y confirmar cambios

Posteriormente le damos en guardar , en el botón sabe al lado superior derecho del mapa (Ver Figura 5).

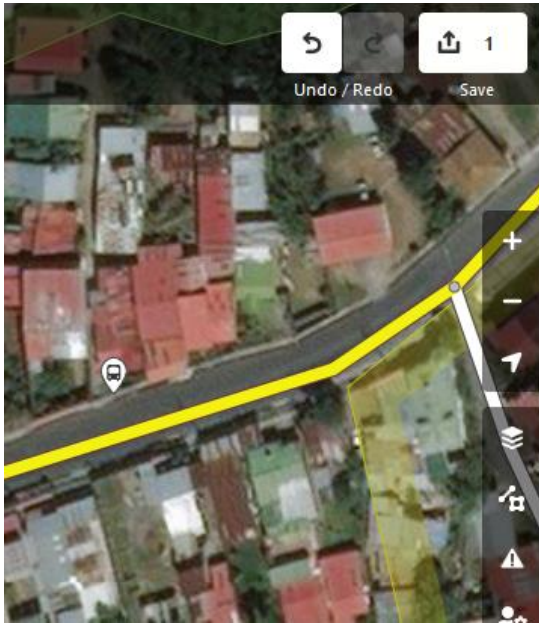


Figura 5. Guardar cambios

Y es así de sencillo como podemos colaborar con el proyecto y también acceder a la información posteriormente que hayamos ingresado. Cabe mencionar que todos estos datos pasan por un estudio previo y en un lapso determinado los cambios se irán haciendo visibles de manera gradual, es posible que si se realizan muchos cambios algunos de ellos no sean aceptados o simplemente se vean reflejados días después. Cuando los cambios hayan sido aceptados podremos acceder y visualizar aquellos puntos que agregamos (Ver Figura 6)



Figura 6. Parada del Bus del Carmen que fue agregada como parte del ejercicio

## 3 LEAFLET

### 3.1 ¿QUÉ ES LEAFLET?

Leaflet es una librería de código abierto basada en JavaScript, esta librería es usada para construir aplicaciones web de mapas. Su lanzamiento data del año 2011, su creador es Vladimir Agafonkin. Esta librería es una de las más populares y usada en sitios como FourSquare, Pinterest y Flickr, es soportada por la mayoría de los dispositivos móviles y plataformas de escritorio que soporten HTML5 T CSS3.

Esta herramienta, permite a los desarrolladores manipular mapas de manera fácil y controlada, contiene muchas funcionalidades como cambiar el estilo del mapa (Tiles), agregar marcadores, personalizar marcadores con íconos, popups y tooltips, entre otros. Además permite el uso de WMS, GeoJSON layers y Vector Layers así como otros plugins para distintas tareas como lo son Routine Machine y Marker Cluster, los cuales veremos en secciones posteriores.[3]

### 3.2 PRIMEROS PASOS EN LEAFLET

- Primero creamos una carpeta donde guardaremos los archivos que crearemos, posteriormente creamos un archivo dentro de dicha carpeta llamado index.html y construimos la estructura básica de html. (Ver Figura 7)

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Document</title>
</head>
<body>

</body>
</html>
```

Figura 7. Estructura básica html

- Segundo agregamos en el head la importación del CSS y JavaScript de Leaflet. (Ver Figura 8)

```
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
```

```

<meta http-equiv="X-UA-
Compatible" content="ie=edge">
<!--
Archivo CSS debe ir antes del JavaScript-->
<link rel="stylesheet" href="https://unpkg.com/leaflet@1.6.0/dist/leaflet.css"
integrity="sha512-
xwE/Az9zrjBIPhAcBb3F6JVqxf46+CDLwFLMH1o
Nu6KEQCAWi6HcDUbeOfBIptF7tcCzusKFjFw2yu
vEpDL9wQ=="
crossorigin=""/>
<!--
Archivo JavaScript debe ir después del CSS-->
<script src="https://unpkg.com/leaflet@1.6.0/dist/leaflet.js"
integrity="sha512-
gZwIG9x3wUXg2hdXF6+rVKLF/0Vi9U8D2Ntg4Ga
5I5BZpVKVx1JWbSQtxPSiUTtC0TjtG0mxa1AJPu
V0CPthew=="
crossorigin=""></script>
<title>Document</title>
</head>

```

Figura 8. Agregando leaflet

- Tercero creamos un elemento div dentro del body y colocamos un identificador en este caso mapid. (Ver Figura 9)

```

<body>
  <div id = "mapid"></div>
</body>

```

Figura 9. Creando el div donde estará el mapa

- Cuarto creamos un archivo style.css dentro de la carpeta, donde estableceremos el tamaño del mapa (Ver Figura 10)

```

html, body {
  height: 100%;
}
#mapid {
  height: 100%;
}

```

Figura 10. Creando el tamaño del mapa

- Quinto creamos un archivo llamado index.js dentro de la carpeta, y en el creamos el mapa y agregamos un estilo o piel dentro de alguna función. (Ver Figura 11)

```

function map(){
  var mymap = L.map('mapid').setView(
    [51.505, -0.09], 13);
  L.tileLayer('https://api.mapbox.com/
styles/v1/{id}/tiles/{z}/{x}/{y}?acces
s_token={accessToken}', {
    attribution: 'Map data &copy; <a hr
ef="https://www.openstreetmap.org/">Ope
nStreetMap</a> contributors, <a href="h
ttps://creativecommons.org/licenses/by-
sa/2.0/">CC-BY-
SA</a>, Imagery © <a href="https://www.
mapbox.com/">Mapbox</a>',
    maxZoom: 18,
    id: 'mapbox/streets-v11',
    accessToken: 'your.mapbox.access.to
ken'
  }).addTo(mymap);
}

```

Figura 11. Creando el mapa en JavaScript

- Sexto, agregamos al head de nuestro index.html el archivo de styles.css y index.js. (Ver Figura 12).

```

<link rel="stylesheet" type="text/css"
href="style.css">
<script src="index.js"></script>

```

Figura 12. Agregando index.js y styles.css

- En el body de nuestro index.html agregamos el onload = "nombre de la función", esto para que la página cargue el mapa almacenado en el JavaScript que creamos. (Ver Figura 13)

```

<body onload="map()">

```

Figura 13. Agregando función onload



- Sétimo, simplemente abrimos nuestro index en cualquier navegador y veremos desplegado nuestro mapa en leaflet. (Ver Figura 14)

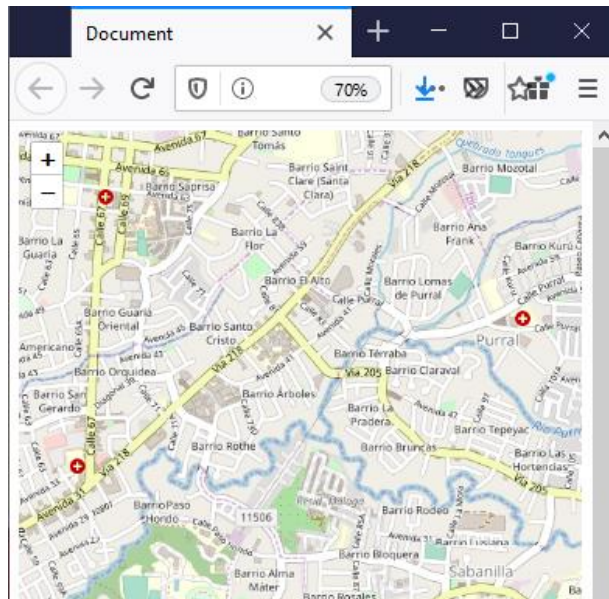


Figura 14. Nuestro primer mapa en leaflet

### 3.3 AGREGANDO MARKERS

Leaflet nos permite agregar marcadores y personalizarlos a nuestro gusto, para agregar un marcador basta con escribir una línea de código. (Ver Figura 15)

```
var marker = L.marker(coordinates).addTo(mymap);
```

Figura 15. Creando un marcador

Para agregar un popup o un tooltip simplemente, después de crear el marcador le agregamos esta funcionalidad con dos líneas de código (Ver Figura 16)

```
marker.bindPopup("Contenido popup");
marker.bindTooltip("Contenido tooltip");
```

Figura 16. Agregando popup y tooltip

Algo muy interesante que mencionamos en reiteradas ocasiones es la posibilidad que nos ofrece leaflet de personalizar nuestros iconos, para ello, antes de crear el marcador simplemente creamos nuestro icono (Ver Figura 17).

```
var myicon = L.icon({
  iconUrl: 'imagen.png',
```

```
iconSize: [50, 50],
iconAnchor: [22, 44],
popupAnchor: [-3, -50]
});
```

Figura 17. Creando un ícono

Posteriormente cuando se crea el marcador, agregamos el icono creado a dicha línea (Ver Figura 18)

```
var marker = L.marker(coordinates,{icon:myicon}).addTo(mymap);
```

Figura 18. Agregando ícono a mi marcador

De esta manera, creamos nuestro marcador de manera individual de manera rápida y fácil, lo que nos permite ocuparnos más de la lógica que debamos desarrollar para resolver algún problema. Por ejemplo, si deseáramos crear más de un marcador una buena solución sería crear un archivo JSON que contuviera las coordenadas de todos los puntos que deseamos marcar en nuestro mapa, no nos olvidemos que Leaflet hace uso de OpenStreetMap y que por tanto podríamos usar los puntos que ingresáramos de ciertas rutas desde OSM para poder construir el archivo con coordenadas. Luego simplemente podemos obtener los datos desde el JSON y con un arreglo de objetos de coordenadas obtenidas de dicho documento, simplemente creamos un ciclo que vaya posicionando cada marcador en el mapa haciendo uso de la línea para creación de un marker.

### 3.4 ¿QUÉ ES ROUTING MACHINE?

Es un plugin que se puede agregar a leaflet que nos permite de manera fácil dibujar la ruta de los puntos que deseamos. Al igual que leaflet, es una herramienta fácil de usar, entender y muy flexible. Basta con unas pocas líneas de código y podremos obtener la ruta demarcada, así como las direcciones para llegar a cada uno de los puntos. Pueden usarse varios puntos intermedios para crear una guía sobre una ruta específica que se desee construir o simplemente darle un punto inicial y final, dándole la libertad a la herramienta de encontrar el camino por sí sola.[4]

### 3.5 PRIMEROS PASOS EN ROUTING MACHINE

Como en cada una de las herramientas que hemos utilizado hasta el momento, es necesario incluir el CSS y JS de Routing machine en el header de un archivo html (Ver Figura 18).

```
<link rel="stylesheet" href="https://unpkg.com/leaflet@1.2.0/dist/leaflet.css" />
```

```
<link rel="stylesheet" href="https://unpkg.com/leaflet-routing-machine@latest/dist/leaflet-routing-machine.css" />
<script src="https://unpkg.com/leaflet@1.2.0/dist/leaflet.js"></script>
<script src="https://unpkg.com/leaflet-routing-machine@latest/dist/leaflet-routing-machine.js"></script>
```

Figura 19. Agregando CSS Y JS de Routing Machine

Es importante destacar que para hacer uso de la librería es necesario haber creado un mapa y realizado las configuraciones necesarias como previamente lo hicimos en secciones anteriores.

Una vez incluida la librería en tan solo unas líneas de código donde se crea una ruta, se añaden los puntos y se agrega al mapa (Ver Figura 20), esto nos permite obtener una ruta entre dos o más puntos

```
L.Routing.control({
  waypoints: [
    L.latLng(57.74, 11.94),
    L.latLng(57.6792, 11.949)
  ]
}).addTo(map);
```

Figura 20. Agregando puntos para marcar ruta

### 3.6 RESULTADO USANDO LEAFLET, MARKERS Y ROUTING MACHINE

En la siguiente imagen se puede apreciar los marcadores sobre la ruta que sigue el bus del Carmen de Guadalupe en sentido Carmen- San José haciendo uso de coordenadas obtenidas en OSM producto de la edición del mapa y posteriormente realizando la programación en leaflet haciendo uso de los markers y dibujando la ruta con routing machine. (Ver Figura 21)

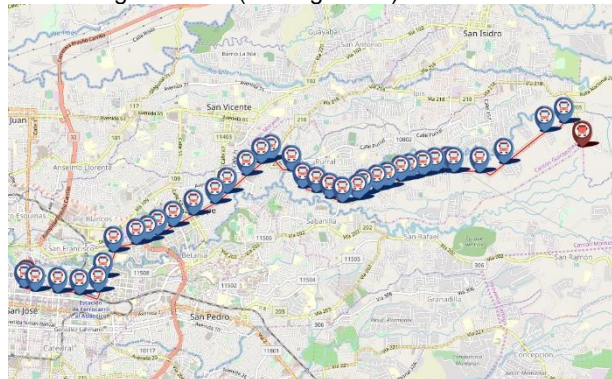


Figura 21. Parada de la ruta Carmen San José

### 3.7 ¿QUÉ SON LOS MARKER CLUSTER?

Los Marker Cluster son un plugin creado por Dave Leaver, un colaborador de Leaflet. Básicamente, encontró una problemática con el uso de marcadores, cuando tienden a ser muchos marcadores el mapa se vuelve un desastre y unos marcadores caen encima de otros y logrando que el mapa se vea desordenado y muy poblado. Por lo que se le ocurrió la grandiosa idea de crear los Marker Cluster, los cuales permiten agrupar marcadores e incluso personalizar cada uno de ellos. Para ello, simplemente basta asignar la cantidad de hijos que tendrá cada cluster, y crear tantos clusters como agrupaciones de deseen. [5]

### 3.8 PRIMEROS PASOS EN CLUSTER MARKER

Como con las anteriores herramienta o plugins, es importante incluir el archivo CSS y JS de los Marker Clusters, para ello simplemente agregamos los links y scripts (Ver Figura 22) en el header, y también es importante reiterar que para su correcto funcionamiento es necesario tener un archivo html previamente configurado con un mapa de leaflet para poder hacer uso de los marker cluster.

```
<link rel="stylesheet" href="https://unpkg.com/leaflet.markercluster@1.3.0/dist/MarkerCluster.css" />
<link rel="stylesheet" href="https://unpkg.com/leaflet.markercluster@1.3.0/dist/MarkerCluster.Default.css" />
<script src="https://unpkg.com/leaflet@1.3.1/dist/leaflet.js" integrity="sha512-NsX9X4HebavoBvEbuyp3I7od5tA0UzAxs+j83KgC8PU0kgB4XiK4Lfe4y4cgBtaRjQJEFcw+oC506aPT2L1zw==" crossorigin=""></script>
<script src="https://unpkg.com/leaflet.markercluster@1.3.0/dist/leaflet.markercluster.js"></script>
```

Figura 22. Agregando CSS Y JS de Marker Cluster

Ahora, simplemente basta crear tantos clusters como agrupaciones se deseen haciendo uso de solamente tres líneas de código (Ver Figura 23).

```
var markers = new L.MarkerClusterGroup(
);

markers.addLayer(L.marker([175.3107, -
37.7784]));

map.addLayer(markers);
```

### Figura 23. Agregando Marker Clusters

### 3.9 RESULTADO USANDO MARKER CLUSTER

En la siguiente imagen se puede apreciar un cluster que contiene las paradas de buses en Jaboncillal (Ver Figura 24)



Figura 24. Marker Cluster paradas Jaboncillal

Conforme se hace zoom, estas paradas se van subdividiendo en los distintos barrios que existen en Jaboncillal. En la siguiente imagen se puede apreciar un cluster que contiene las paradas de buses en Jaboncillal (Ver Figura 25)



Figura 25. Despliegue de paradas en barrios de Jaboncillal

Hasta llegar a cada parada en específico donde podremos incluso apreciar que cada parada contiene un ícono diferente. (Ver Figura 26)



Figura 26. Paradas específicas en Jaboncillal

## 4 REFERENCIAS

- [1] OpenStreetMap. (2019) Obtenido: 13 de enero 2020, desde <https://es.wikipedia.org/wiki/OpenStreetMap>
- [2] OpenStreetMap. Obtenido: 13 de enero 2020, desde <https://www.openstreetmap.org/about>
- [3] Leaflet (software). (2019). Obtenido: 13 de enero 2020, desde [https://en.wikipedia.org/wiki/Leaflet\\_\(software\)](https://en.wikipedia.org/wiki/Leaflet_(software))
- [4] Leaflet Routing Machine. Retrieved 13 January 2020, from <https://www.liedman.net/leaflet-routing-machine/>
- [5] Leaver, D. (2012). Leaflet.MarkerCluster 0.1 Released - Leaflet - a JavaScript library for interactive maps. Obtenido: 13 de enero 2020, desde <https://leafletjs.com/2012/08/20/guest-post-markercluster-0-1-released.html>