

Introdução às tecnologias Web - ITW

Aula 11 – Maps & Charts

Sumário

Introdução às API's

Google Maps

OpenStreet Maps

LeafLetJS

Google Charts

ChartJS



Revisão:

O que é uma API? Para que serve?

Uma API (acrónimo do termo em inglês *Application programming interface*) consiste na especificação de um conjunto de rotinas e estruturas de dados estabelecidos por um desenvolvedor de software.

Através da utilização dessas rotinas e dessas estruturas de dados cria-se uma interface que permite a utilização dessas funcionalidades por aplicativos que não pretendem envolver-se nos detalhes da implementação do software, mas apenas usar seus serviços.

Google Maps

A Google disponibiliza várias APIs para permitir o acesso programático aos seus mais diversos serviços.

Uma delas é o “Google Maps” que permite a pesquisa e visualização de mapas.

São disponibilizadas duas APIs principais para aceder a este serviço:

Google Maps Embed API

<https://developers.google.com/maps/documentation/embed/guide>

Google Maps JavaScript API v3

<https://developers.google.com/maps/documentation/javascript/tutorial>

Google Maps

Embed API

A Embed API do serviço Google Maps utiliza um pedido simples, em HTTP, para devolver um mapa.

Este mapa é dinâmico e interativo e pode facilmente ser inserido numa página web, atribuindo a URL da API ao atributo "src" de um elemento "iframe".

A sintaxe para a URL da API é:

`https://www.google.com/maps/embed/v1/MODO?key=CHAVE&PARÂMETROS`

MODO = "place" | "directions" | "search" | "view"

CHAVE = chave gratuita de acesso à API

PARÂMETROS = parâmetros específicos de cada modo ou parâmetros adicionais e opcionais

Google Maps

Embed API – Modo “Place”

MODO = “**place**” – mapa de um lugar

PARÂMETROS =

Obrigatórios

q – especifica o lugar a pesquisar

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <title>Google Maps | Place</title>
  <link href="https://fonts.googleapis.com/css?family=Open+Sans:300" rel="stylesheet">
  <link href="../../Content/bootstrap.min.css" rel="stylesheet" />
  <link href="../../Content/font-awesome.css" rel="stylesheet" />
  <style>
    body { font-family: 'Open Sans', sans-serif; }
    .myFrame { width: 100%; height: 450px; border: none; }
  </style>
</head>
<body>
  <div class="container">
    <div class="page-header">
      <h2>Google Maps | Place</h2>
      q=Forum Aveiro, Aveiro, PT
    </div>
    <iframe class="myFrame"
      src="https://www.google.com/maps/embed/v1/place?key=.....
      &q=Forum Aveiro, Aveiro, PT"></iframe>
    </div>
    <script src="../../Scripts/jquery-3.6.0.min.js"></script></body>
</html>
```

Google Maps | Place

q=Forum Aveiro, Aveiro, PT

Forum Aveiro
R. do Batalhão de Caçadores, 3810-064 Aveiro, Portugal
4,3 ★★★★★ 571 comentários
[Ver mapa maior](#)

[Direções](#) [Guardar](#)

[Iniciar sessão](#)

Map labels: Jardim do Rossio, R. do Clube dos Galitos, Museu da Cidade de Aveiro, Sport Zone, Fórum Aveiro, Sé Catedral de Aveiro ou Igreja de São Domingos, Museu de Aveiro (Sta. Joana), Parque de Santo António, Centro Ciência Viva de Aveiro, Santa Joana, Av. 25 de Abril, Escola Secundária José Estêvão, Meliá Ria Hotel & Spa, Ecori Turis, R. Dr. Lourenço Peixinho, R. Cmte. Rocha e C, R. José Afonso, R. Carlos Aleluia, R. Jaime Moniz.

©2016 Google - Dados do mapa ©2016 Google Termos de Utilização Comunicar um erro no mapa

```
<iframe class="myFrame"
src="https://www.google.com/maps/embed/v1/place?key=(...)
&q=Forum Aveiro, Aveiro, PT"></iframe>
```


Google Maps

Embed API – Modo “Directions”

MODO = “**directions**” – permite a representação de um mapa com um trajeto definido entre um conjunto de pontos

PARÂMETROS =

Obrigatórios

origin – origem do trajeto

destination – destino do trajeto

Opcionais

waypoints – pontos de passagem no trajeto

mode – modo de viajar (*driving, walking, bicycling, transit, flying*)

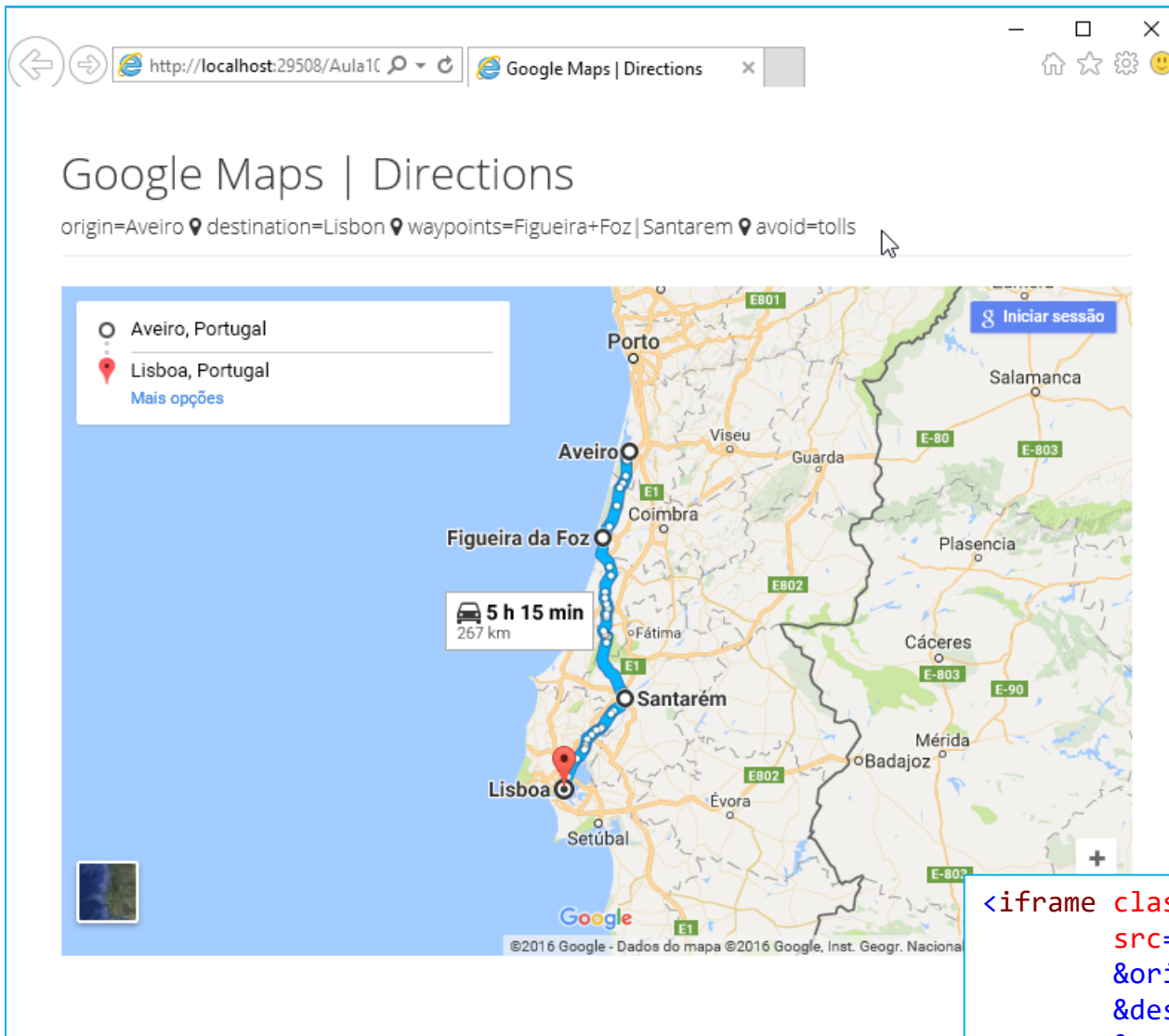
avoid – questões a evitar (*tolls, highways*)

units – unidades de medida do trajeto (*metric, imperial*)

```

<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <title>Google Maps | Directions</title>
  <link href="https://fonts.googleapis.com/css?family=Open+Sans:300" rel="stylesheet">
  <link href="../../Content/bootstrap.min.css" rel="stylesheet" />
  <link href="../../Content/font-awesome.css" rel="stylesheet" />
  <style>
    body { font-family: 'Open Sans', sans-serif; }
    .myFrame { width: 100%; height: 450px; border: none; }
  </style>
</head>
<body>
  <div class="container">
    <div class="border-bottom pt-4 pb-2 mb-3">
      <h2>Google Maps | Directions</h2>
      origin=Aveiro -x- destination=Lisbon -x- waypoints=Figueira+Foz|Santarem -x- avoid=tolls
    </div>
    <iframe class="myFrame"
      src="https://www.google.com/maps/embed/v1/directions?key=....."
      &origin=Aveiro
      &destination=Lisbon
      &waypoints=Figueira+Foz|Santarem
      &avoid=tolls"
      allowfullscreen></iframe>
    </div>
    <script src="../../Scripts/jquery-3.6.0.min.js"></script></body>
</html>

```



```
<iframe class="myFrame"
src="https://www.google.com/maps/embed/v1/directions?key=(...)
&origin=Aveiro
&destination=Lisbon
&waypoints=Figueira+Foz|Santarem
&avoid=tolls"
allowfullscreen></iframe>
```

Google Maps

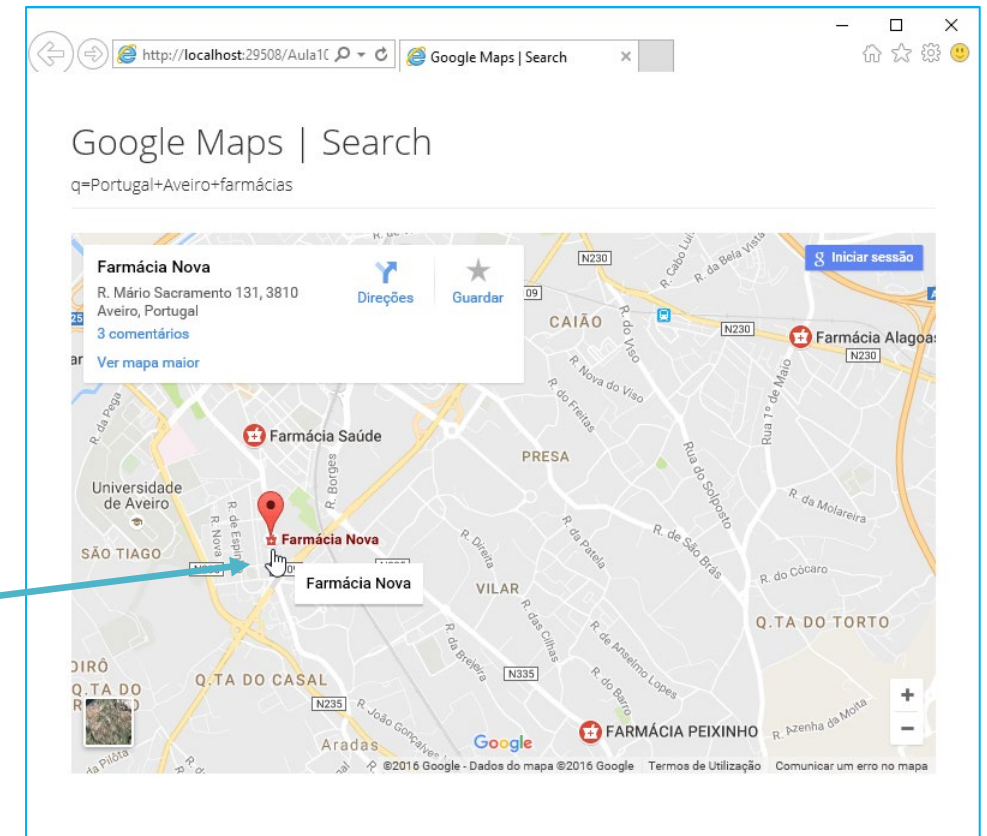
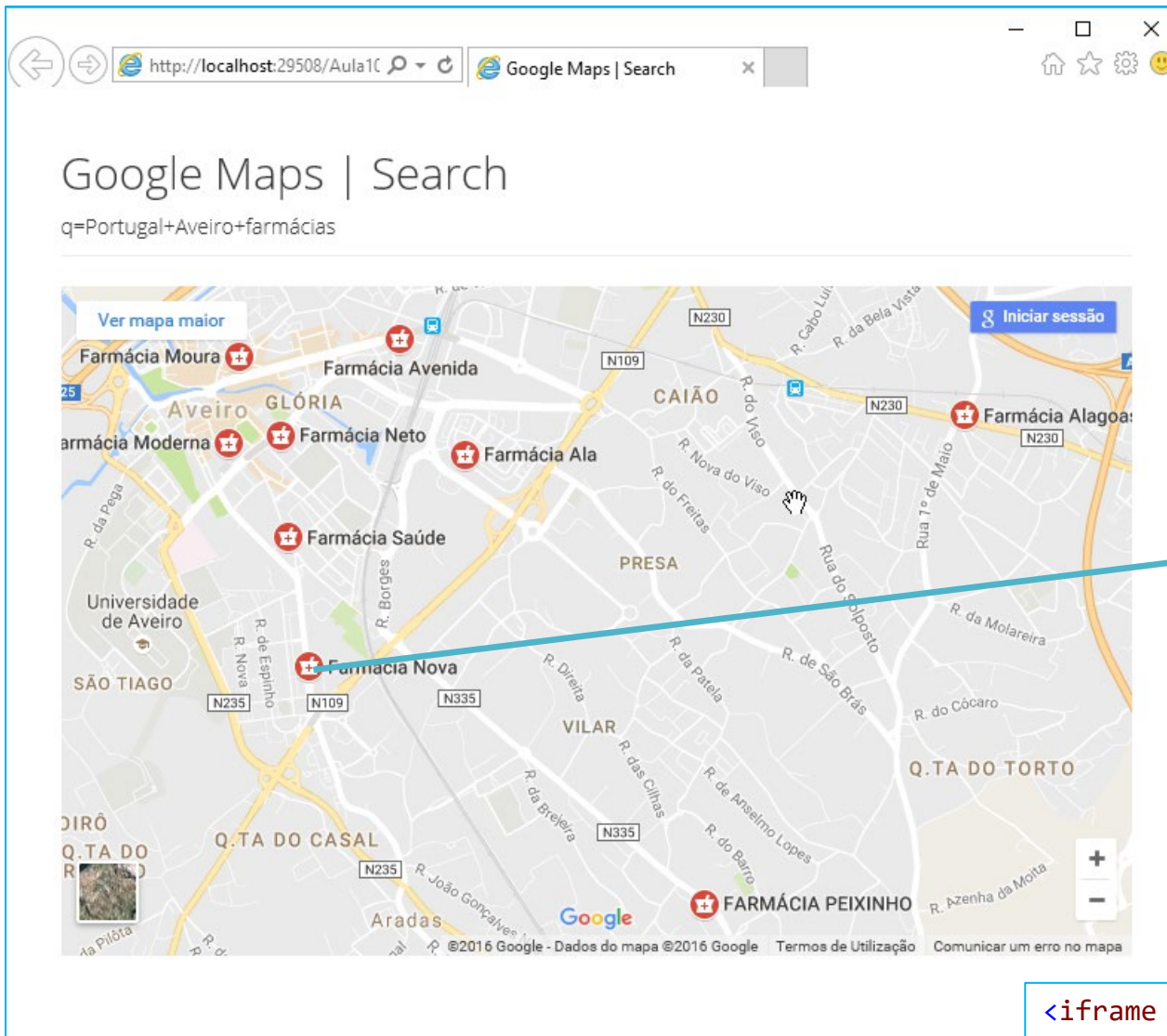
Embed API – Modo “Search”

MODO = “**search**” – permite a representação de um mapa com pesquisa de características num lugar (livrarias, farmácias, restaurantes, ...)

PARAMETROS =

q – especifica o lugar e as características a pesquisar

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <title>Google Maps | Search</title>
  <link href="https://fonts.googleapis.com/css?family=Open+Sans:300" rel="stylesheet">
  <link href="../../Content/bootstrap.min.css" rel="stylesheet" />
  <link href="../../Content/font-awesome.css" rel="stylesheet" />
  <style>
    body {font-family:'Open Sans', sans-serif}
    .myFrame { width: 100%; height: 450px; border:none; }
  </style>
</head>
<body>
  <div class="container">
    <div class="border-bottom pt-4 pb-2 mb-3">
      <h2>Google Maps | Search</h2>
      q=Portugal+Aveiro+farmácias
    </div>
    <iframe class="myFrame"
      src="https://www.google.com/maps/embed/v1/search?key=.....
      &q=Portugal+Aveiro+farmácias"
      allowfullscreen></iframe>
    </div>
    <script src="../../Scripts/jquery-3.6.0.min.js"></script>
  </body>
</html>
```



```
<iframe class="myFrame"
src="https://www.google.com/maps/embed/v1/search?key=(...)
&q=Portugal+Aveiro+farmácias"
allowfullscreen></iframe>
```

Google Maps

Embed API – Modo “View”

MODO = “**view**” – permite desenhar um mapa com centro num par de coordenadas GPS

PARÂMETROS:

Obrigatório

center – par de coordenadas GPS do centro do mapa sepearadas por vírgula

Opcionais

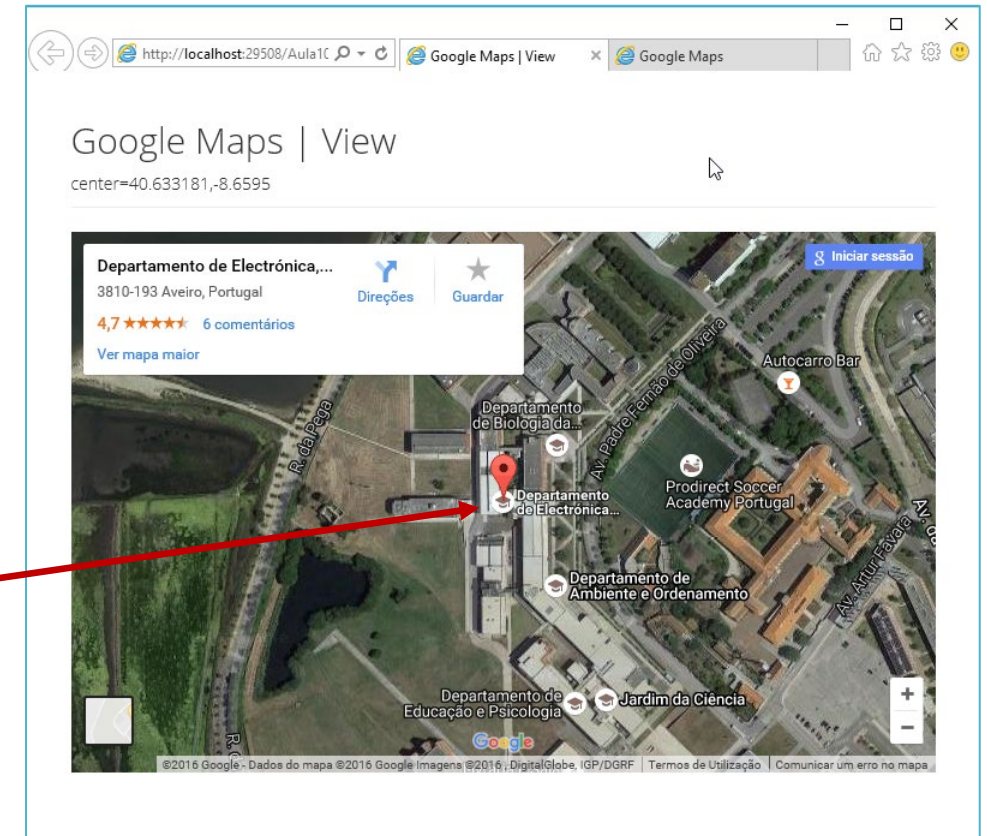
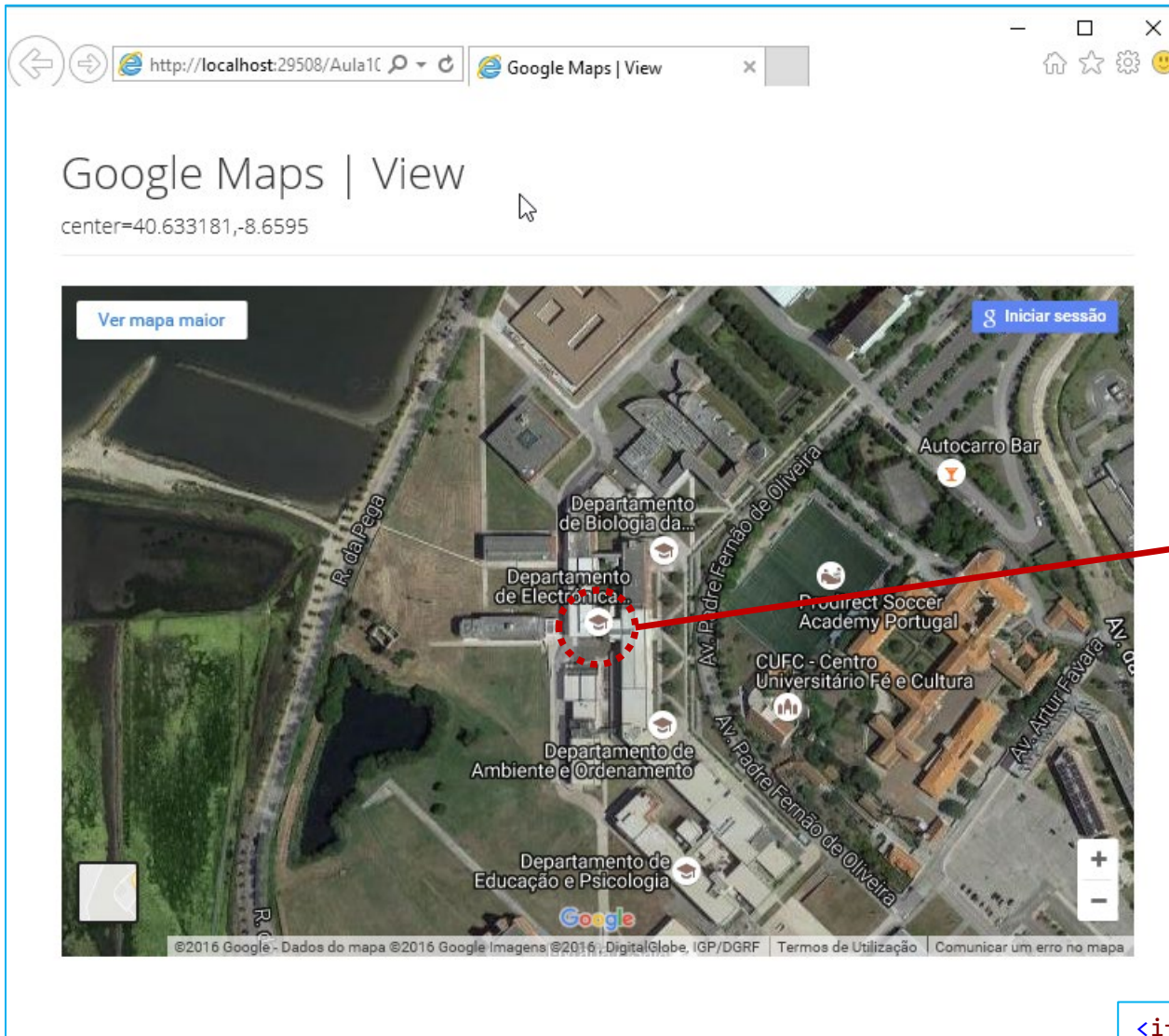
zoom – nível de ampliação do mapa (0 - 21)

maptype – tipo de mapa (roadmap, satellite)

language – linguagem a usar nos elementos da interface

region – mostra os limites apropriados, com base na sensibilidade geo-políticas (ccTLD – cc = código da região)

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <title>Google Maps | View</title>
  <link href="https://fonts.googleapis.com/css?family=Open+Sans:300" rel="stylesheet">
  <link href="../../Content/bootstrap.min.css" rel="stylesheet" />
  <link href="../../Content/font-awesome.css" rel="stylesheet" />
  <style>
    body {font-family:'Open Sans', sans-serif}
    .myFrame { width: 100%; height: 450px; border:none; }
  </style>
</head>
<body>
  <div class="container">
    <div class="border-bottom pt-4 pb-2 mb-3">
      <h2>Google Maps | View</h2>
      center=40.633181,-8.6595
    </div>
    <iframe class="myFrame"
      src="https://www.google.com/maps/embed/v1/view?key=(...)"
      &center=40.633181,-8.6595
      &zoom=17
      &maptype=satellite"
      allowfullscreen></iframe>
    </div>
    <script src="../../Scripts/jquery-3.6.0.min.js"></script>
  </body>
</html>
```

```
<iframe class="myFrame"
src="https://www.google.com/maps/embed/v1/view?key=(...)
&center=40.633181,-8.6595
&zoom=17
&motype=satellite"
allowfullscreen></iframe>
```

Google Maps

JavaScript API v3

A **JavaScript API v3** do serviço Google Maps também realiza um pedido, em HTTP, para que seja devolvido um mapa dinâmico e interativo.

Que diferenças para a Embed API?

O mapa não necessita de ser colocado dentro de uma "iframe"; pode ser inserido em qualquer elemento HTML;

É possível um grau muito mais elevado de controlo e personalização do mapa;

Exige programação (em javascript → **não funciona em jQuery**).

Para funcionar em jQuery, é necessário instalar de mais uma livreria...

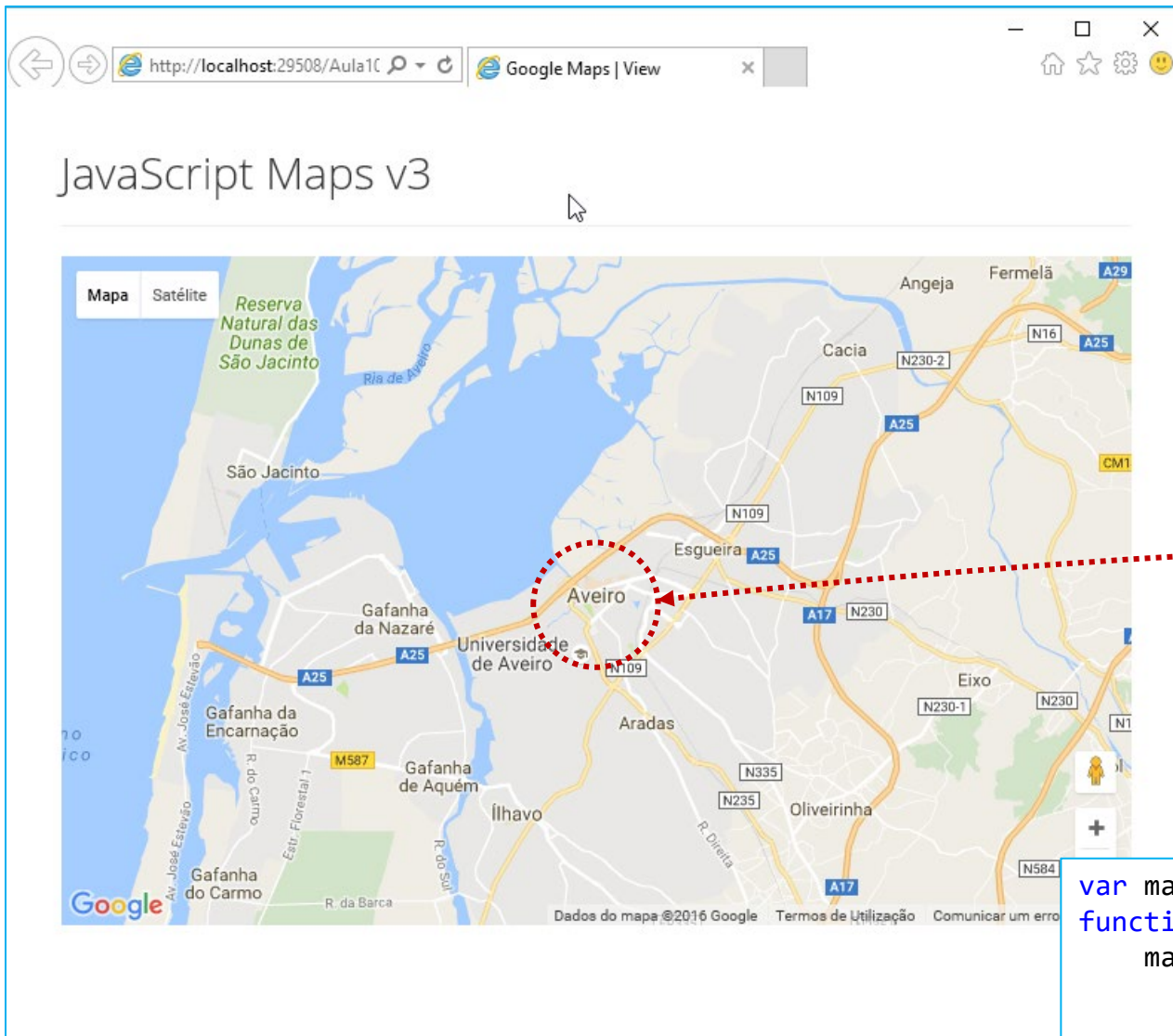
```

<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <title>Google Maps | View</title>
  <link href="https://fonts.googleapis.com/css?family=Open+Sans:300" rel="stylesheet">
  <link href="../../Content/bootstrap.min.css" rel="stylesheet" />
  <link href="../../Content/font-awesome.css" rel="stylesheet" />
  <style>
    body { font-family:'Open Sans', sans-serif; height: 100%; margin: 0; padding: 0; }
    #map { width: 100%; height: 450px; border:none; }
  </style>
</head>
<body>
  <div class="container">
    <div class="border-bottom pt-4 pb-2 mb-3">
      <h2>JavaScript Maps v3</h2>
    </div>
    <div id="map"></div>
  </div>

  <!-- Não temos jQuery ! -->
  <script async defer src="https://maps.googleapis.com/maps/api/js?key=(...)M&callback=initMap"></script>
  <script type="text/javascript">
    var map;
    function initMap() {
      map = new google.maps.Map(document.getElementById('map'), {
        center: { lat: 40.6413266, lng: -8.6536320999999982 },
        zoom: 12
      });
    }
  </script>
</body>
</html>

```





```
var map;  
function initMap() {  
    map = new google.maps.Map(document.getElementById('map'), {  
        center: { lat: 40.6413266, lng: -8.653632099999982 },  
        zoom: 12  
    });  
}
```


Utilização da localização atual do utilizador para sugestão de serviços ...

A interface **Geolocation** é um objeto capaz de obter, programaticamente, a posição do dispositivo. Ele dá à página Web acesso ao local do dispositivo.

Isso permite que um site ou aplicativo ofereçam resultados personalizados com base na localização do utilizador.

Métodos da interface **Geolocation**:

Geolocation.getCurrentPosition()

*Determina a localização atual do dispositivo e devolve um objeto **Position** com os dados.*

Geolocation.watchPosition()

*Retorna um **Long** que representa **callback funtion** que deve ser invocada sempre que a localização do dispositivo é alterada.*

Geolocation.clearWatch()

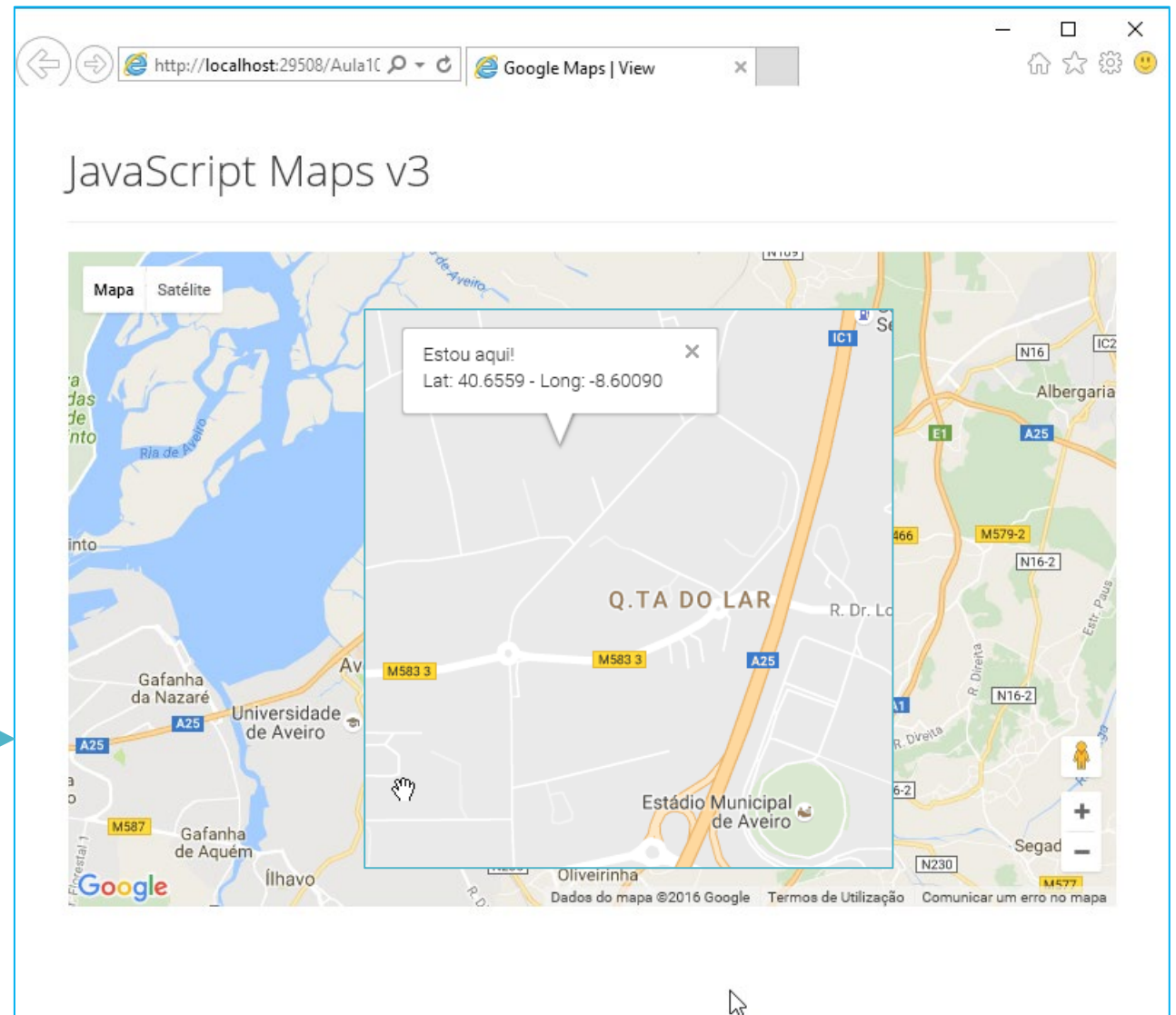
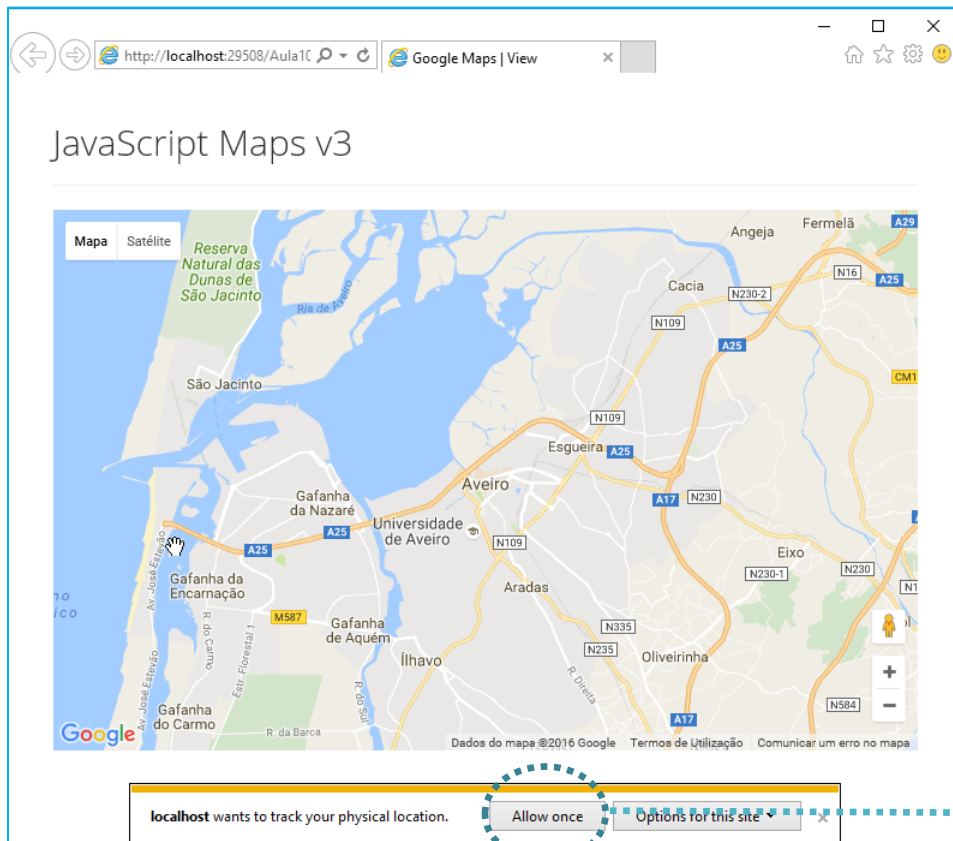
*Remove o **handler** para a função instalada usando o método **watchPosition()**.*

Para +info: <https://developer.mozilla.org/en-US/docs/Web/API/Geolocation>

```

<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <title>Google Maps | View</title>
  <link href="../Content/bootstrap.min.css" rel="stylesheet" />
  <style>
    #map { width: 100%; height: 450px; border:none; }
  </style>
</head>
<body>
  <div class="container">
    <div class="border-bottom pt-4 pb-2 mb-3">
      <h2>JavaScript Maps v3</h2>
    </div>
    <div class="container">
      <div id="map"></div>
    </div>
  </div>
  <script src="../Scripts/jquery-3.6.0.min.js"></script>
  <script async defer src="https://maps.googleapis.com/maps/api/js?key=(...)&callback=initMap"></script>
  <script type="text/javascript">
    var map;
    function initMap() {
      map = new google.maps.Map(document.getElementById('map'), {
        center: { lat: 40.6413266, lng: -8.653632099999982 },
        zoom: 12
      });
      if (navigator.geolocation) {
        navigator.geolocation.getCurrentPosition(
          function (position) {
            var pos = new google.maps.LatLng(position.coords.latitude, position.coords.longitude);
            var infowindow = new google.maps.InfoWindow(
              {
                map: map, position: pos,
                content: 'Estou aqui!<br/>Lat: ' + pos.lat().toString().substr(0, 8) + ' - Long: ' + pos.lng().toString().substr(0, 8)
              }
            );
            map.setCenter(pos);
          },
          function () {
            handleNoGeolocation(true);
          }
        );
      }
    }
  </script>
</body>
</html>

```



OpenStreetMaps

OpenStreetMaps

OpenStreetMap (OSM) é um projeto de mapeamento colaborativo para criar um mapa livre e editável do mundo.

É construído por uma comunidade de mapeadores que contribuem e mantêm dados sobre estradas, trilhos, cafés, estações ferroviárias e muito mais, em todo o mundo.

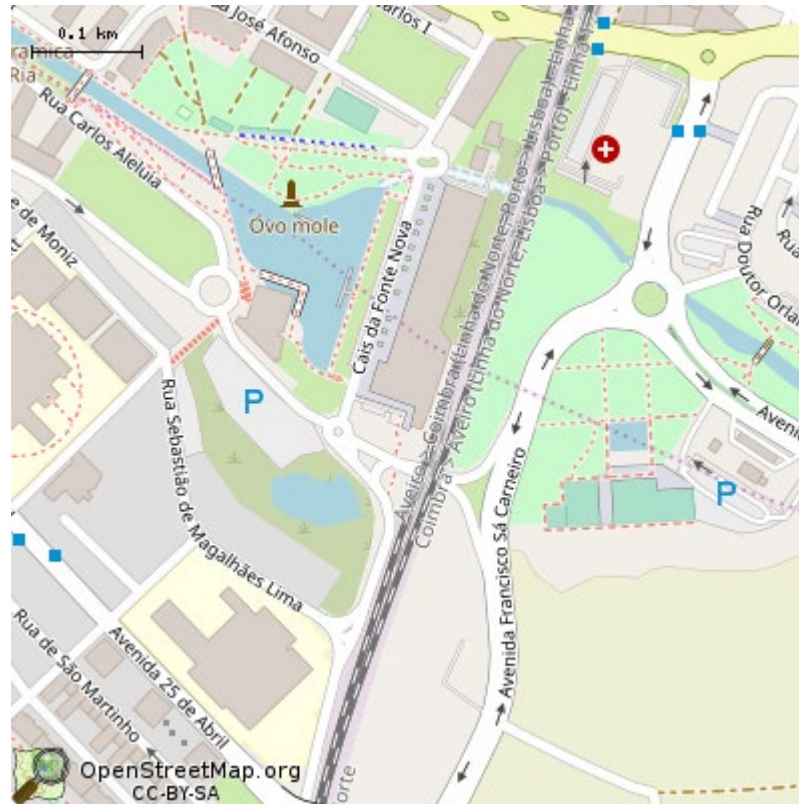
Há uma página/comunidade dedicada especificamente a Portugal.

<https://wiki.openstreetmap.org/wiki/Portugal>

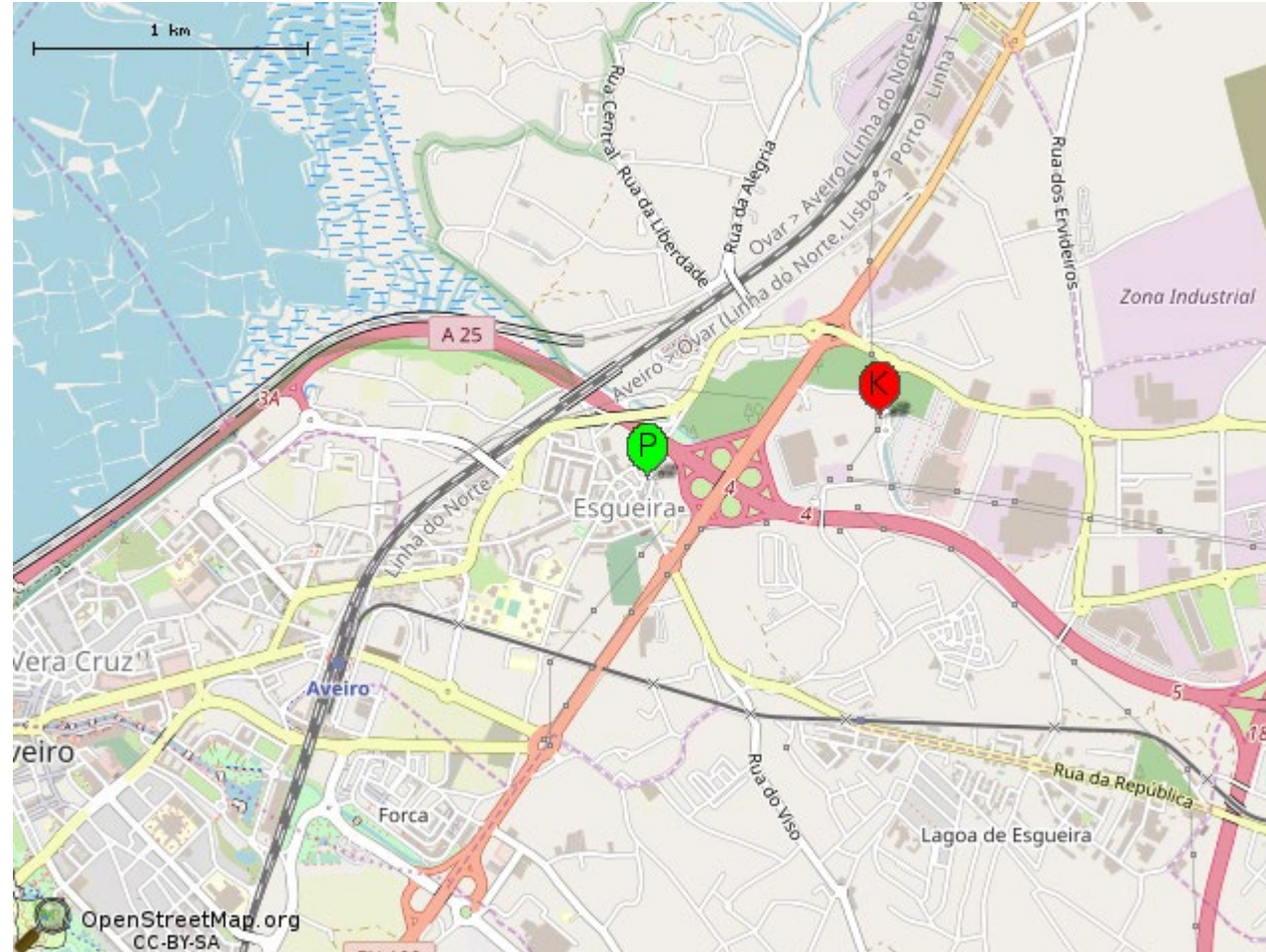


```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8" />
  <title></title>
  <link href="../../../Content/bootstrap-reboot.min.css" rel="stylesheet" />
</head>
<body>
  <iframe width="800" height="480" frameborder="0" scrolling="no" marginheight="0" marginwidth="0"
    src="https://www.openstreetmap.org/export/embed.html?bbox=-8.662963807582857%2C40.63158680486605%2C-
8.656767904758455%2C40.634114924693606&layer=mapnik&marker=40.63285291226819%2C-8.659527897834778"
    style="border: 1px solid black"></iframe><br />
  <small><a href="https://www.openstreetmap.org/?mlat=40.63285&mlon=-8.65953#map=19/40.63285/-8.65953"
target="_blank">View Larger Map</a></small>
</body>
</html>
```

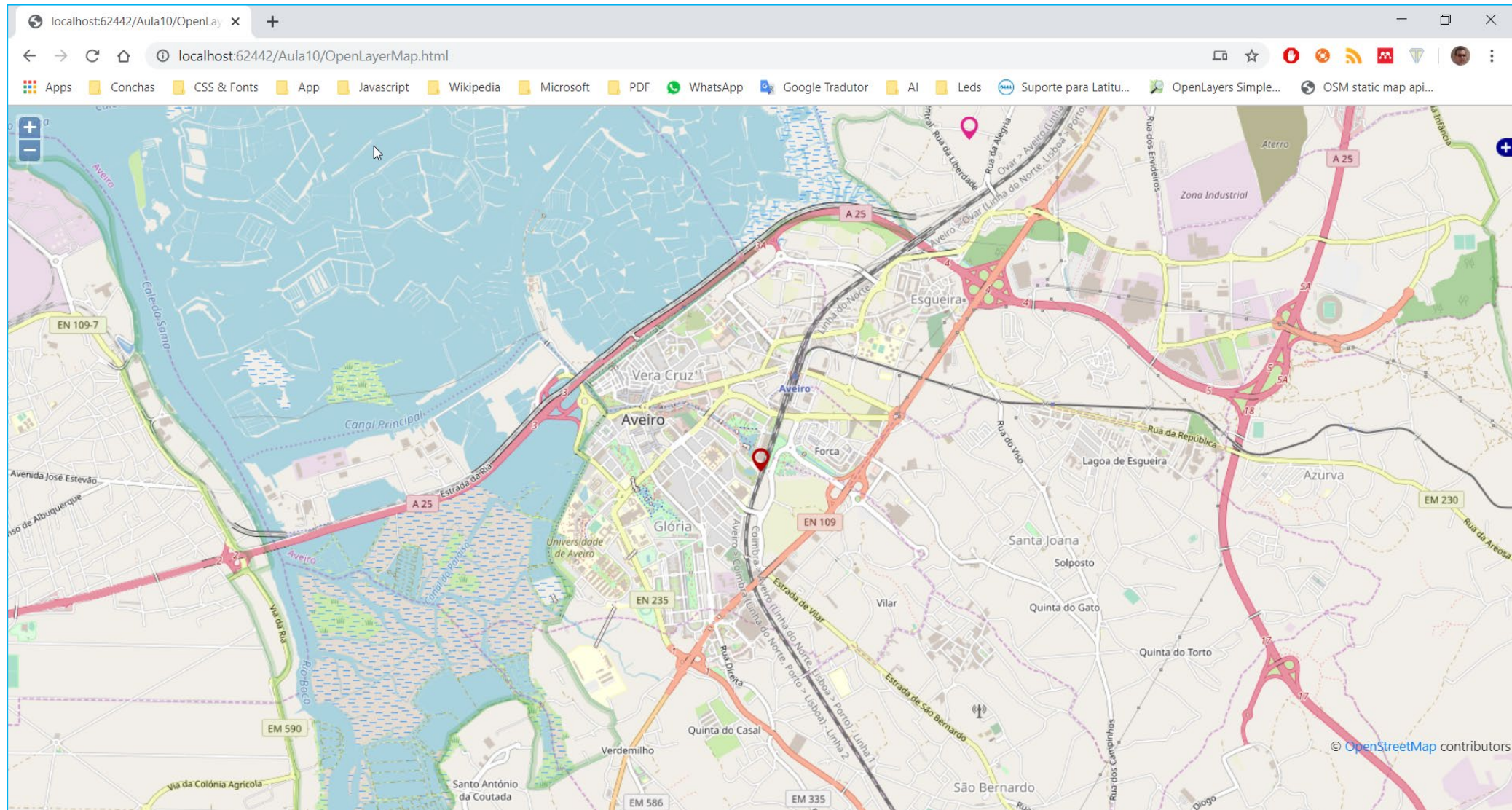
Outro provider mas ... os mesmos mapas



<http://geomap.nagvis.org/?module=map¢er=-8.6434525,40.637786&zoom=16&width=400&height=400>



<http://geomap.nagvis.org/?module=map¢er=-8.627084,40.649439&zoom=14&type=mapnik&width=640&height=480&points=-8.627084,40.649439,pointImagePattern:greenP;-8.617084,40.651439,pointImagePattern:redK>



```

<html>
<head>
  <link href="../../Content/bootstrap.min.css" rel="stylesheet" />
</head>
<body>
  <div id="map"></div>
  <script src="../../Scripts/bootstrap.min.js"></script>
  <script src="http://www.openlayers.org/api/OpenLayers.js"></script>
  <script>
    map = new OpenLayers.Map("map");
    map.addLayer(new OpenLayers.Layer.OSM());

    var pois = new OpenLayers.Layer.Text("Points of Interest - POIS",
    {
      location: "./POIS.txt",
      projection: map.displayProjection
    });
    map.addLayer(pois);
    // create layer switcher widget in top right corner of map.
    var layer_switcher = new OpenLayers.Control.LayerSwitcher({});
    map.addControl(layer_switcher);
    //Set start centrepoint and zoom
    var lonLat = new OpenLayers.LonLat(-8.6434525, 40.637786)
      .transform(
        new OpenLayers.Projection("EPSG:4326"), // transform from WGS 1984
        map.getProjectionObject() // to Spherical Mercator Projection
      );
    var zoom = 14;
    map.setCenter(lonLat, zoom);

  </script>
</body>
</html>

```

POIS.TXT

lat	lon	title	description	icon	iconSize	iconOffset
40.637786	-8.6434525	CMA	Câmara Municipal de Aveiro	/images/marker_red.png	24,24	-12,-12
40.6699851	-8.6582032	Title Two	Description two	/images/marker_blue.png	24,24	-12,-12

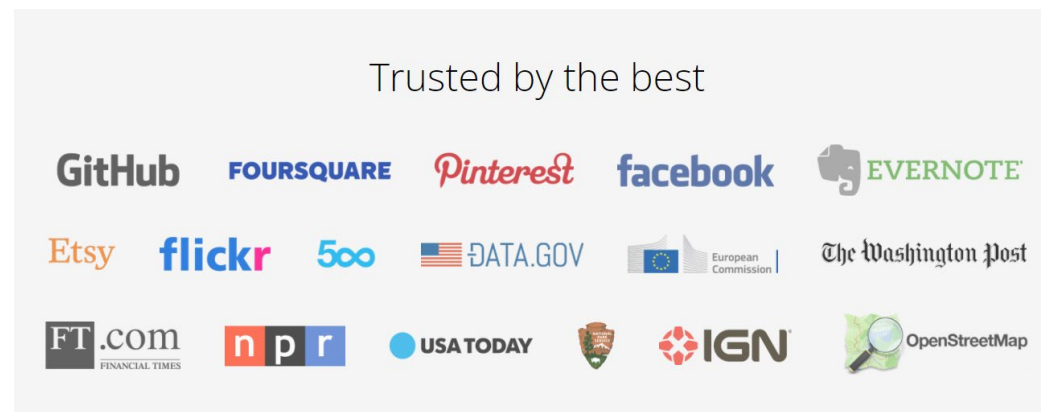
A biblioteca leafletJS

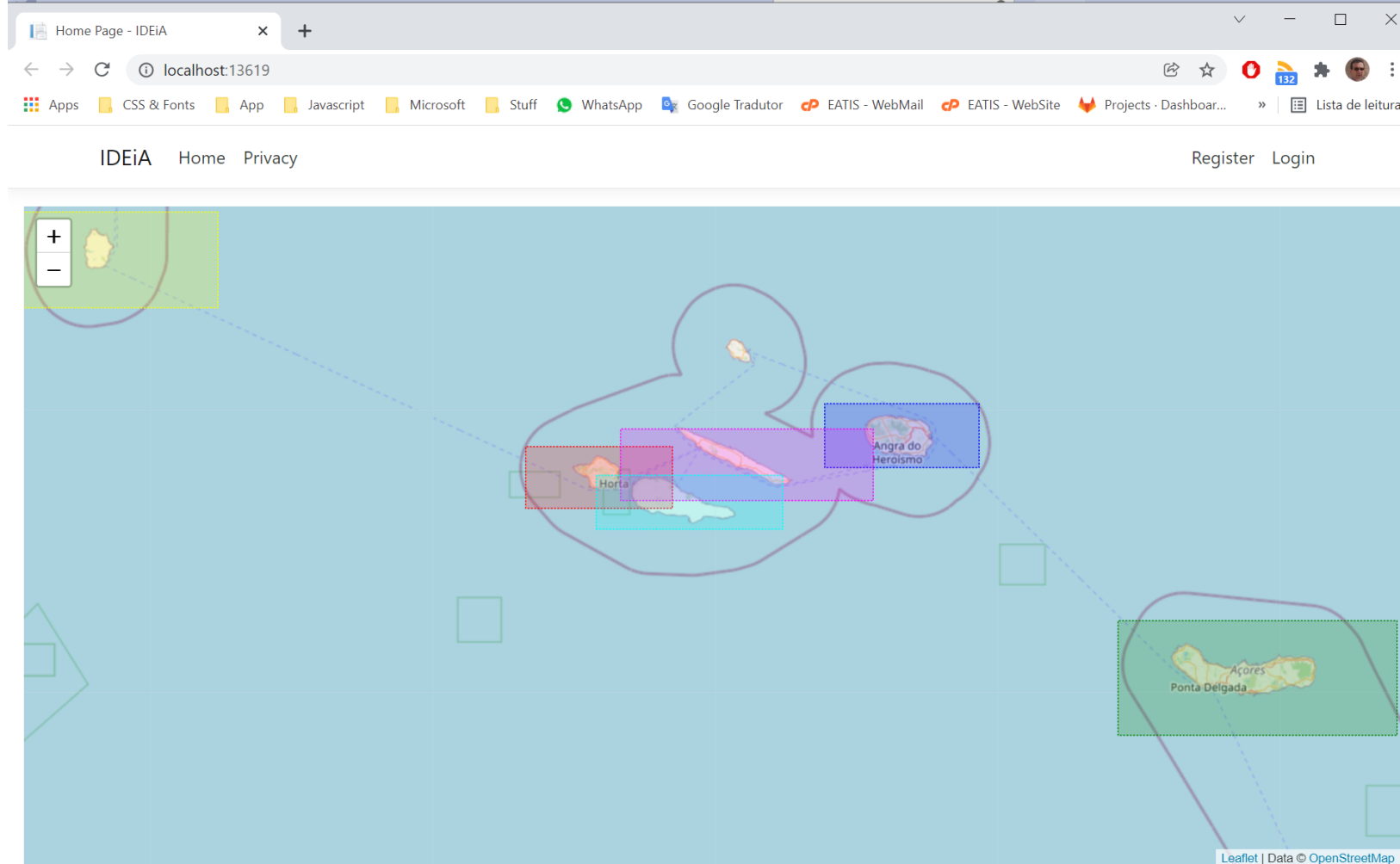
<https://leafletjs.com/> - Sep 4, 2020 — Leaflet 1.7.1

Leaflet é a biblioteca JavaScript de código aberto líder para mapas interativos compatíveis com dispositivos móveis.

Tem cerca de 39 KB de JS possuindo os recursos de mapeamento que a maioria dos desenvolvedores precisa.

Funciona com eficiência em todas as principais plataformas de desktop e móveis, pode ser estendido com vários plug-ins, está bem documentada e possui um código-fonte simples e legível.





Açudes - Região Autónoma dos Açores (RAA)

Informação referente aos açudes existentes nas ilhas da RAA (Faial, Flores, Terceira, Pico, São Jorge e São Miguel). Cartografia vetorial produzida à escala 1: 25 000. A produção da presente informação teve início nos estudos de base do Plano Regional da Água dos Açores, sendo revista em 2018 no âmbito da elaboração do Relatório de Estado das Ribeiras dos Açores (RERA).. Os dados geográficos do PRA foram obtidos através da recolha e processamento de um significativo conjunto de informação georeferenciada, incluindo elementos e bases cartográficas provenientes do IGeoE e do Plano Regional do Ordenamento do Território dos Açores (PROTA).

```

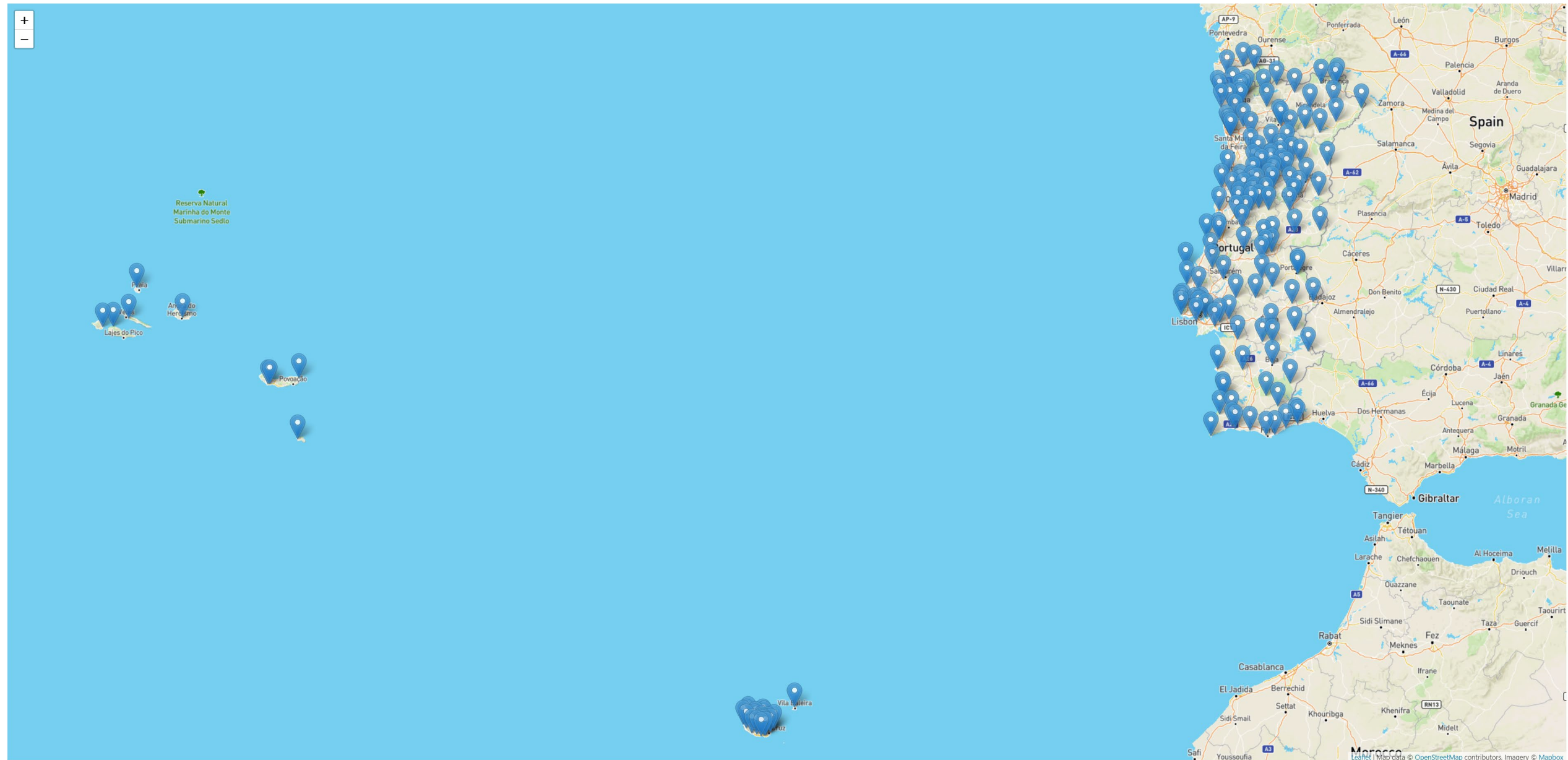
<script src="~/lib/leaflet/leaflet.min.js"></script>
<script>
$(document).ready(function () {
    var center = [39.726188659668 - (39.726188659668 - 36.927604675293) / 2, -31.2688159942627 + (-25.0131645202637 + 31.2688159942627) / 2];
    var map = L.map('map').setView(center, 8);
    // Set up the OSM layer
    L.tileLayer(
        'https://{s}.tile.openstreetmap.org/{z}/{x}/{y}.png', {
            attribution: 'Data © <a href="http://osm.org/copyright">OpenStreetMap</a>',
            maxZoom: 18
        }).addTo(map);

    L.rectangle([[38.68, -29.07], [38.44, -28.34]], { color: 'red', weight: 1, dashArray: '1, 2' }).on('click', function (e) {
        console.info(e); // Quando se carragar no objeto, mostra o objeto
    }).addTo(map);
    L.rectangle([[39.59, -31.74], [39.22, -30.60]], { color: 'yellow', weight: 1, dashArray: '1, 2' }).on('click', function (e) {
        console.info(e);
    }).addTo(map);
    L.rectangle([[38.85, -27.58], [38.60, -26.81]], { color: 'blue', weight: 1, dashArray: '1, 2' }).on('click', function (e) {
        console.info(e);
    }).addTo(map);
    L.rectangle([[38.00, -26.12], [37.55, -24.73]], { color: 'green', weight: 1, dashArray: '1, 2' }).on('click', function (e) {
        console.info(e);
    }).addTo(map);
    L.rectangle([[38.75, -28.60], [38.47, -27.34]], { color: 'ff00FF', weight: 1, dashArray: '1, 2' }).on('click', function (e) {
        console.info(e);
    }).addTo(map);
    L.rectangle([[38.57, -28.72], [38.36, -27.79]], { color: 'cyan', weight: 1, dashArray: '1, 2' }).on('click', function (e) {
        console.info(e);
    }).addTo(map);

    // Quando se carragar no mapa, mostra as coordenadas
    map.on('click', function (e) {
        var coord = e.latlng;
        console.log("(" + coord.lat + "," + coord.lng + ")");
    });
});
</script>

```

Lista de Estações



```

<script>
$(document).ready(function () {
    $("#mapid").css("height", window.innerHeight - 214);
    $(window).resize(function () {
        $("#mapid").css("height", window.innerHeight - 214);
    });

    var mymap = L.map('mapid').setView([36.75, -17], 6);

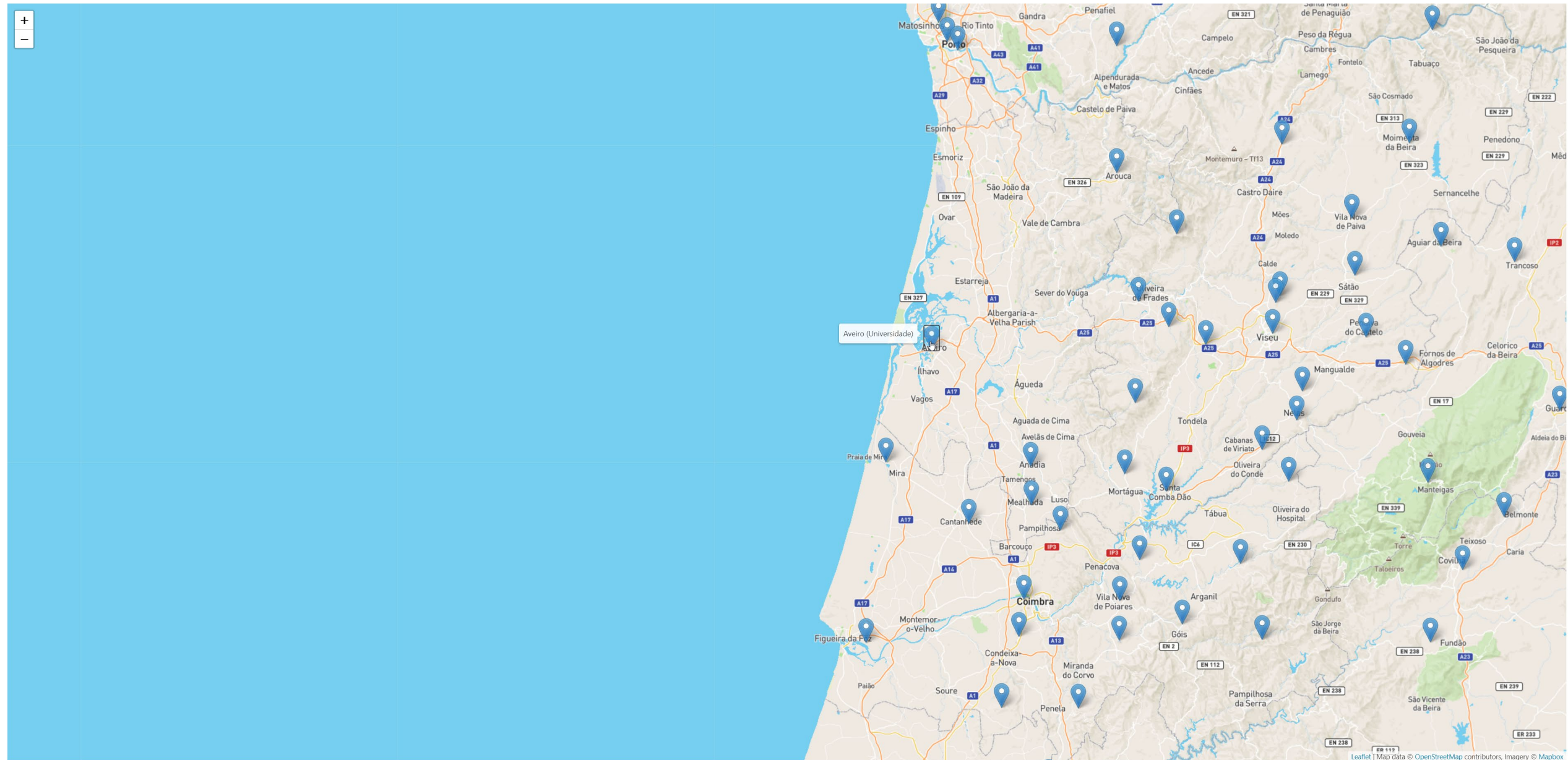
    L.tileLayer('https://api.mapbox.com/styles/v1/{id}/tiles/{z}/{x}/{y}?access_token=...', {
        maxZoom: 18,
        attribution: 'Map data &copy; <a href="https://www.openstreetmap.org/copyright">OpenStreetMap</a> contributors, ' +
            'Imagery &copy; <a href="https://www.mapbox.com/">Mapbox</a>',
        id: 'mapbox/streets-v11',
        tileSize: 512,
        zoomOffset: -1
    }).addTo(mymap);

    //--- Toooooodos os pontos
    L.marker([39.458190000,-31.130100000], { opacity : 0.80})
        .bindTooltip('Flores (Aer&#xF3;dromo)').openTooltip()
        .addTo(mymap);
    //--- (...)
    L.marker([39.466666670,-8.050000000], { opacity : 0.80})
        .bindTooltip('Alvega').openTooltip()
        .addTo(mymap);
    var popup = L.popup();

    mymap.on('click', function (e) {
        popup
            .setLatLng(e.latlng)
            .setContent("You clicked the map at " + e.latlng.toString())
            .openOn(mymap);
    });
});
</script>

```


Lista de Estações



Gráficos

Utilização de gráficos

Os gráficos são frequentemente usados para facilitar a compreensão de grandes quantidades de dados e as relações entre partes dos dados.

Podem ser lidos mais rapidamente do que os dados brutos, por isso são usados numa ampla variedade de cenários.

Podem ser criados à mão (geralmente em papel milimétrico) ou por computador usando um aplicativo de gráficos.

Felizmente, há hoje em dia um conjunto muito alargado de ferramentas que permitem a utilização de gráficos em páginas web.

Há duas abordagens possíveis: a abordagem local, a abordagem remota.

Google Chart API

O que é a Google Chart API?

A Google Chart API é uma ferramenta que permite criar remotamente um gráfico a partir de alguns dados e incorporá-lo numa página da web.

A API do Google Chart cria uma imagem PNG de um gráfico a partir de dados e parâmetros de formatação enviados no pedido HTTP.

São suportados muitos tipos de gráficos e, fazendo a solicitação através de uma tag de imagem, é possível incluir o gráfico numa página da web.

Using Google Charts | C x

← → ↻ ↗ https://developers.google.com/chart/interactive/docs/ ☆ 🔴 📶 🔍 ⚙

Apps Which Programming L Conchas CSS & Fonts Angular2 App Puzzles Phonegap Javascript Wikipedia TAP Portugal check-in Web API docs

Google Charts 🔍 Search ☰ All Products 👤

HOME GUIDES REFERENCE SUPPORT

Overview

Hello, Charts!

Quickstart

Load the Charts Library

Prepare the Data

Customize the Chart

Draw the Chart

Draw Multiple Charts

Chart Types

Chart Gallery

Annotation Charts

Area Charts

Bar Charts

Bubble Charts

Calendar Charts

Candlestick Charts

Column Charts

Combo Charts

Using Google Charts

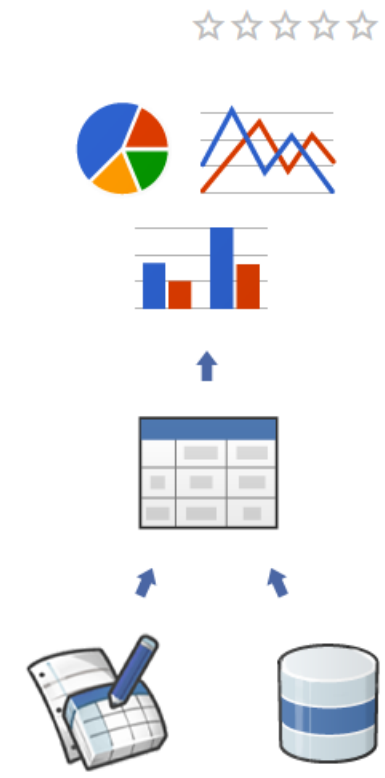
★★★★★

Google Charts provides a perfect way to visualize data on your website. From simple line charts to complex hierarchical tree maps, the [chart gallery](#) provides a large number of ready-to-use chart types.

The most common way to use Google Charts is with simple JavaScript that you embed in your web page. You load some Google Chart libraries, list the data to be charted, select options to customize your chart, and finally create a chart object with an id that you choose. Then, later in the web page, you create a <div> with that id to display the Google Chart.

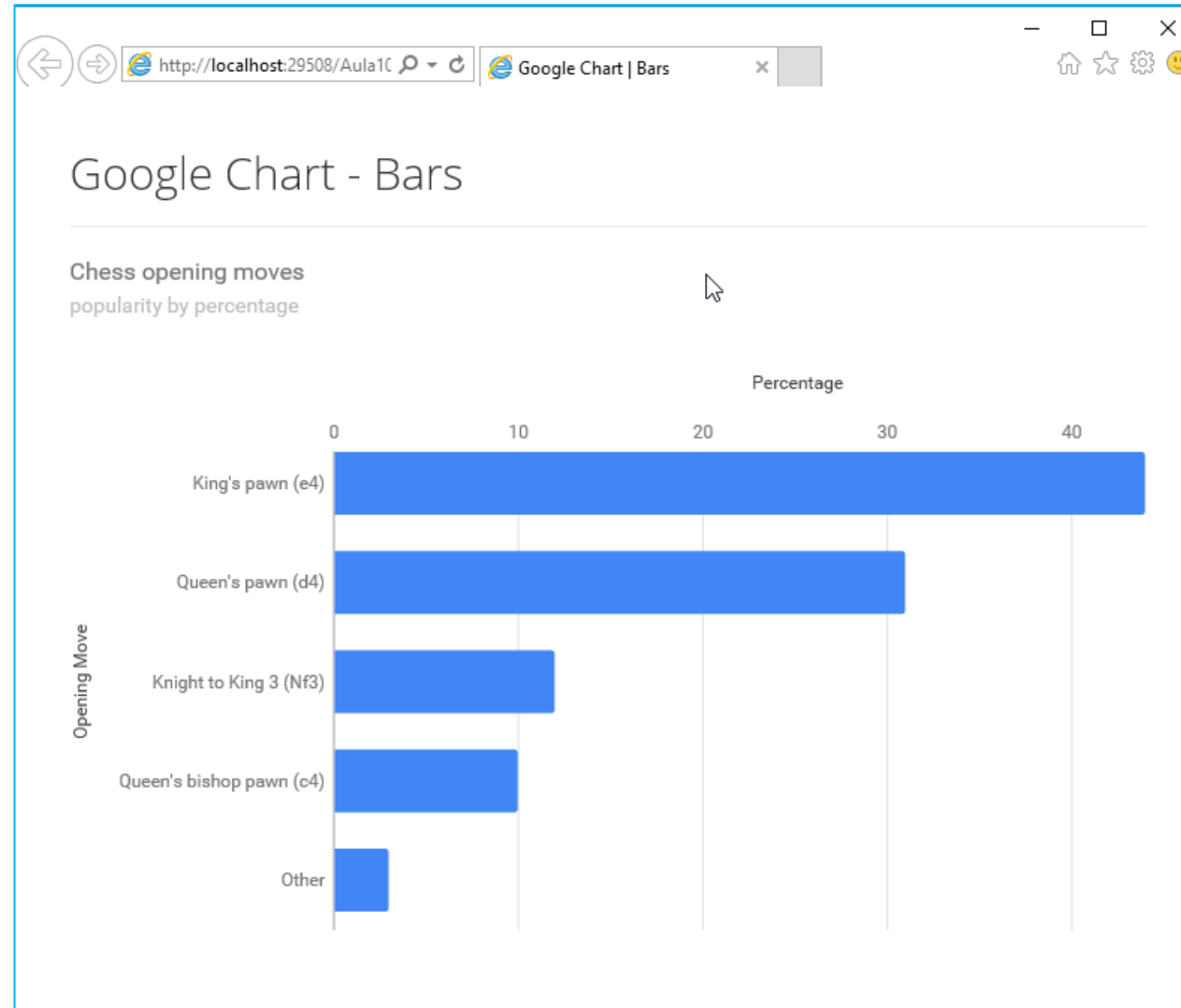
That's all you need to [get started](#).

Charts are exposed as JavaScript classes, and Google Charts provides [many chart types](#) for you to use. The default appearance will usually be all you need, and you can always [customize a chart](#) to fit the look and feel of your website. Charts are highly interactive and expose [events](#) that let you connect them to create complex [dashboards](#) or other experiences [integrated with your webpage](#). Charts are rendered using HTML5/SVG technology to provide cross-browser compatibility (including VML for older IE versions) and cross-platform compatibility (iPhone, iPad, and Android).



The diagram illustrates the workflow of Google Charts. At the bottom, there are two icons: a spreadsheet with a pencil and a database cylinder. Arrows point from both to a table icon. An arrow points from the table to a bar chart icon. Another arrow points from the bar chart to a line chart icon. Finally, an arrow points from the line chart to a pie chart icon. The line and pie charts are grouped together under a five-star rating.

Google Charts - Bar Chart



```

<html>
<head>
  <meta charset="utf-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <title>Google Chart | Bars</title>
  <link href="../Content/bootstrap.min.css" rel="stylesheet" />
  <style>
    #chart { width: 100%; height: 450px; border:none; }
  </style>
</head>
<body>
  <div class="container">
    <div class="border-bottom pt-4 pb-2 mb-3">
      <h2>Google Chart - Bars</h2>
    </div>
    <div id="chart"></div>
  </div>
  <script src="../Scripts/jquery-3.6.0.min.js"></script>
  <script src="https://www.google.com/jsapi" type="text/javascript"></script>

```

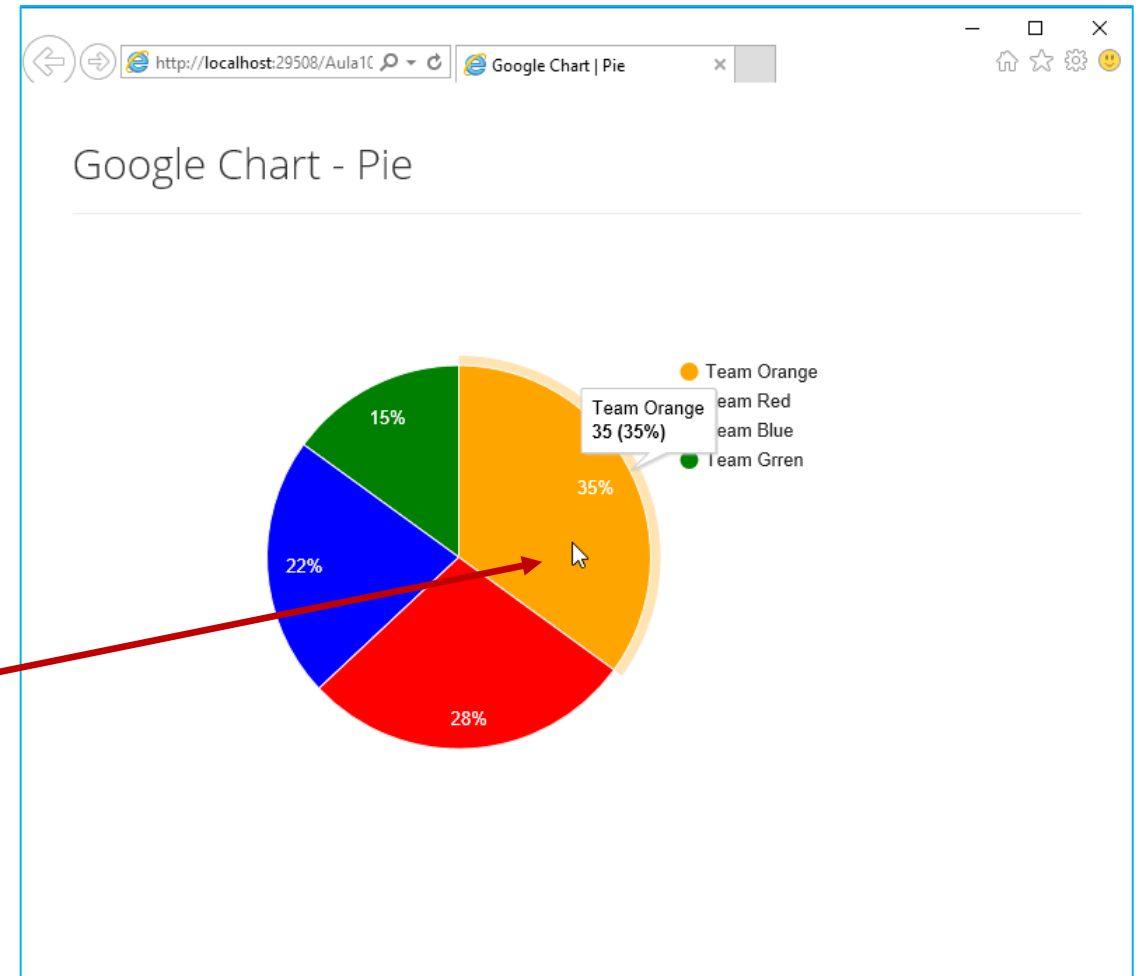
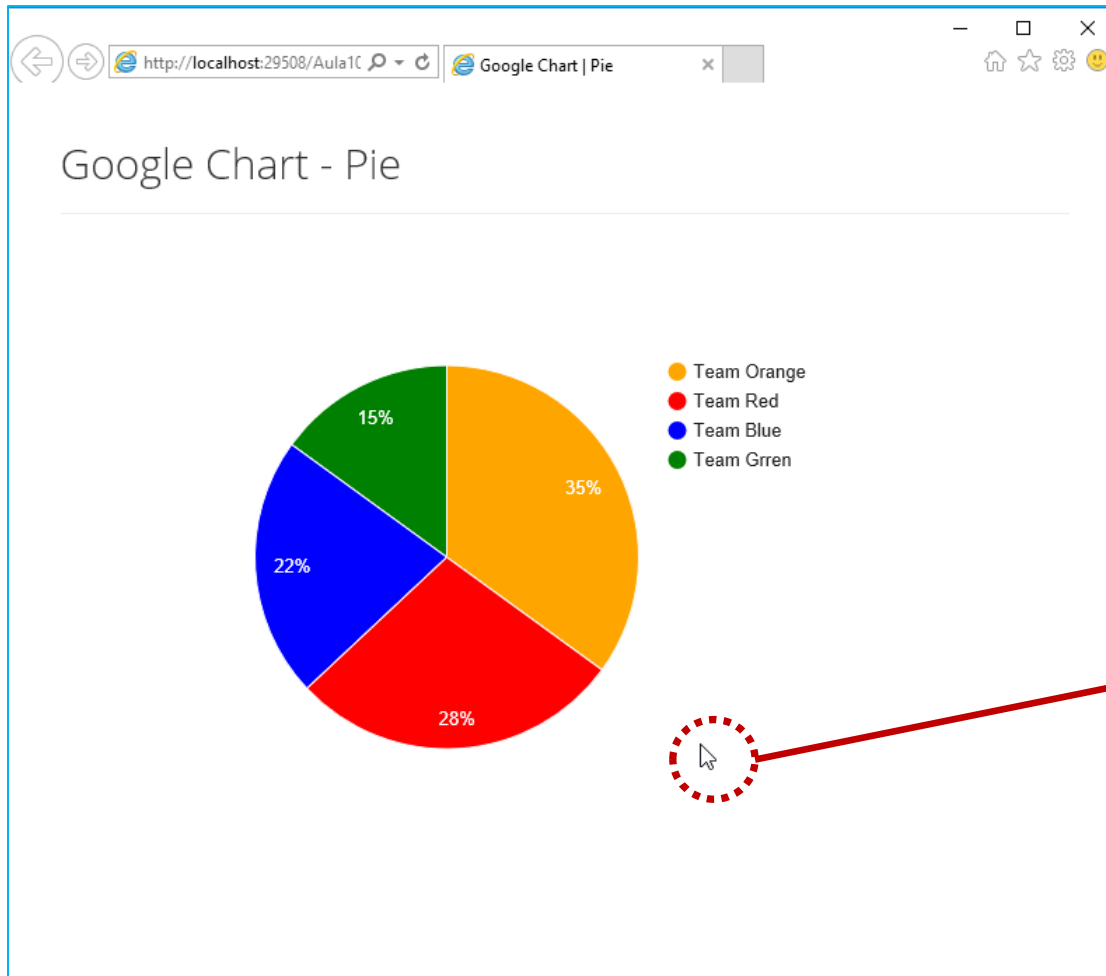
```

<script type="text/javascript">
  google.load("visualization", "1.1", { packages: ["bar"] });
  google.setOnLoadCallback(drawStuff);
  function drawStuff() {
    var data = new google.visualization.arrayToDataTable([
      ['Opening Move', 'Percentage'],
      ["King's pawn (e4)", 44],
      ["Queen's pawn (d4)", 31],
      ["Knight to King 3 (Nf3)", 12],
      ["Queen's bishop pawn (c4)", 10],
      ['Other', 3]
    ]);
    var options = {
      title: 'Chess opening moves',
      width: 800,
      legend: { position: 'none' },
      chart: {
        title: 'Chess opening moves',
        subtitle: 'popularity by percentage'
      },
      bars: 'horizontal', // Required for Material Bar Charts.
      axes: {
        x: {
          0: { side: 'top', label: 'Percentage' } // Top x-axis.
        }
      },
      bar: { groupWidth: "90%" }
    };

    var chart = new google.charts.Bar(document.getElementById('chart'));
    chart.draw(data, options);
  };
</script>
</body>
</html>

```

Google Charts - Pie Chart

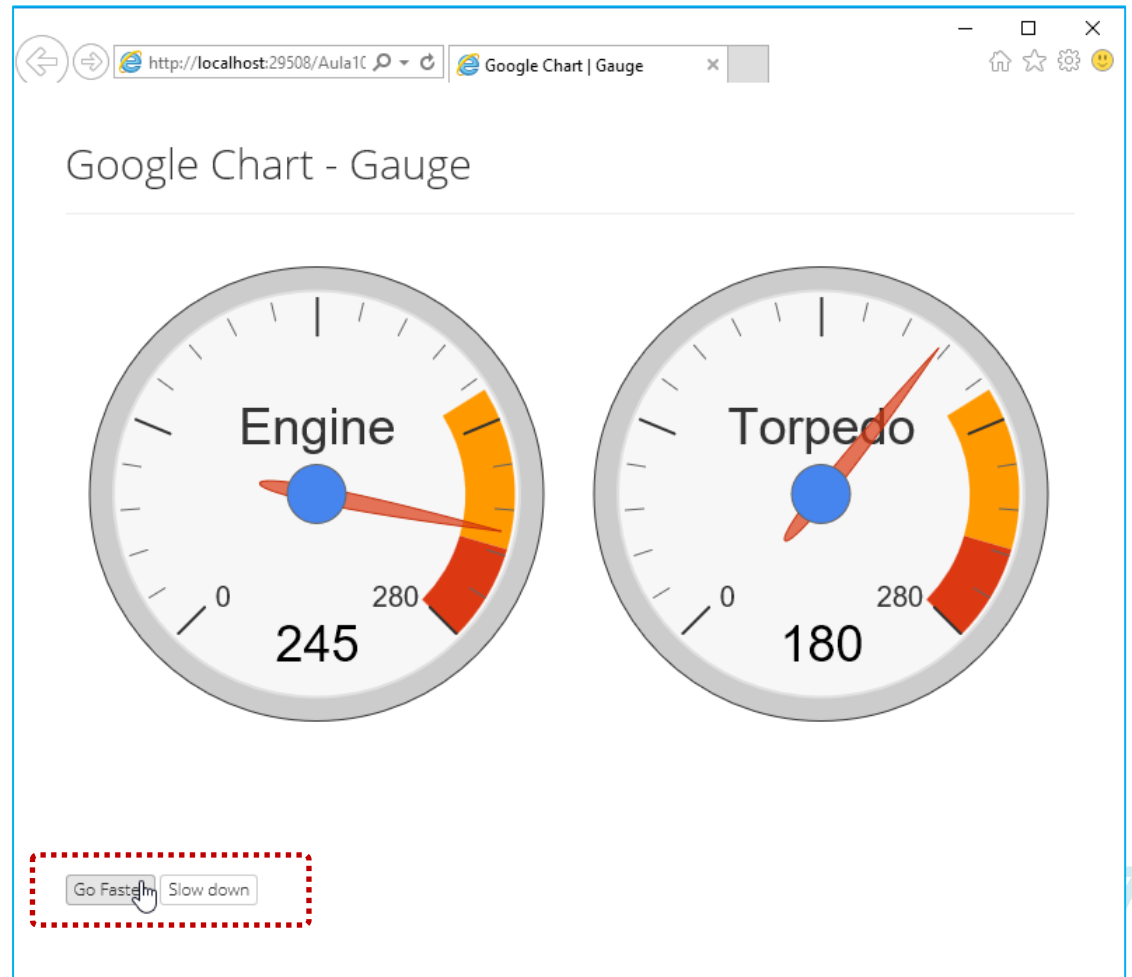
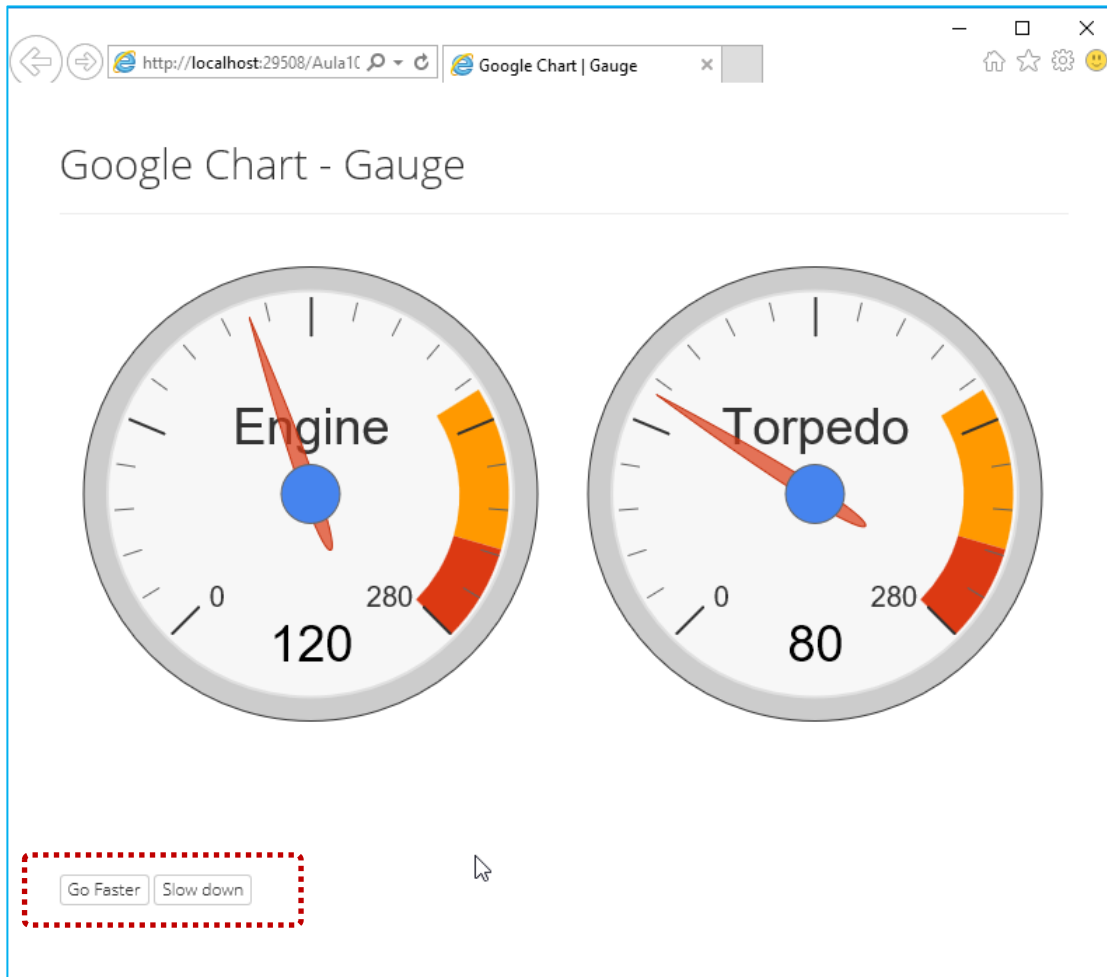


```

<html>
<head>
  <meta charset="utf-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <title>Google Chart | Pie</title>
  <link href="../../Content/bootstrap.min.css" rel="stylesheet" />
  <style>
    #chart { width: 100%; height: 450px; border:none; }
  </style>
</head>
<body>
  <div class="container">
    <div class="border-bottom pt-4 pb-2 mb-3">
      <h2>Google Chart - Pie</h2>
    </div>
    <div id="chart"></div>
  </div>
  <script src="../../Scripts/jquery-3.6.0.min.js"></script>
  <script type="text/javascript" src="https://www.google.com/jsapi"></script>
  <script type="text/javascript">
    google.load("visualization", "1", { packages: ["corechart"] });
    google.setOnLoadCallback(drawChart);
    function drawChart() {
      var data = google.visualization.arrayToDataTable([
        ['Main National Teams', 'Percentage'], ['Team Orange', 35], ['Team Red', 28], ['Team Blue', 22], ['Team Green', 15]
      ]);
      var options = {
        slices: {
          0: { color: 'orange' }, 1: { color: 'red' }, 2: { color: 'blue' }, 3: { color: 'green' }
        }
      };
      var chart = new google.visualization.PieChart(document.getElementById('chart'));
      chart.draw(data, options);
    }
  </script>
</body>
</html>

```

Google Charts - Gauges



```

<body>
  <div class="container">
    <div class="border-bottom pt-4 pb-2 mb-3">
      <h2>Google Chart - Gauge</h2>
    </div>
    <div id="chart"></div>
    <input type="button" value="Go Faster" class="btn btn-default btn-xs" onclick="changeTemp(1)" />
    <input type="button" value="Slow down" class="btn btn-default btn-xs" onclick="changeTemp(-1)" />
  </div>
  <script src="../../Scripts/jquery-3.6.0.min.js"></script>
  <script src="https://www.google.com/jsapi?autoload={ 'modules': ['visualization', 'version': '1', 'packages': ['gauge'] ] }"></script>
  <script type="text/javascript">
    google.load('visualization', '1', { packages: ['gauge'] });
    google.setOnLoadCallback(drawGauge);

    var gauge;
    var gaugeOptions = {
      min: 0, max: 280, yellowFrom: 200, yellowTo: 250, redFrom: 250, redTo: 280, minorTicks: 5
    };

    function drawGauge() {
      gaugeData = new google.visualization.DataTable();
      gaugeData.addColumn('number', 'Engine');
      gaugeData.addColumn('number', 'Torpedo');
      gaugeData.addRows(2);
      gaugeData.setCell(0, 0, 120);
      gaugeData.setCell(0, 1, 80);

      gauge = new google.visualization.Gauge(document.getElementById('chart'));
      gauge.draw(gaugeData, gaugeOptions);
    }

    function changeTemp(dir) {
      gaugeData.setValue(0, 0, gaugeData.getValue(0, 0) + dir * 25);
      gaugeData.setValue(0, 1, gaugeData.getValue(0, 1) + dir * 20);
      gauge.draw(gaugeData, gaugeOptions);
    }
  </script>
</body>

```


ChartJS

Chart.JS

A grande vantagem de utilização de uma livraria local reside no fato de os dados ficarem sempre do nosso lado;

ao enviar os dados para o Google, nunca poderemos afirmar com total segurança, o que será feito com eles.

Outra vantagem, reside na possibilidade de implementar gráficos mais complexos e que saiam do comum.

Apresentarei alguns exemplos

Instalando o Chart.JS localmente

The screenshot shows the NuGet Package Manager interface. On the left, a list of packages is displayed under the 'Browse' tab. The search filter is set to 'chart.js'. The first package, 'Chart.js' by Nick Downie, is highlighted with a red dashed box. It has 835K downloads and version v2.9.3. Below it are other packages like 'ChartJS', 'Chart.Mvc', 'ChartJSCore', 'chart.js.TypeScript.DefinitelyTyped', and 'ChartJs.Blazor'. On the right, the details for the selected 'Chart.js' package are shown. It includes the package icon, name, version (2.9.3), and buttons for 'Uninstall' and 'Update'. The 'Options' section is expanded, showing the package description: 'Simple HTML5 Charts using the canvas element.' It also lists the author (Nick Downie), license (View License), date published (Thursday, November 28, 2019), project URL (http://www.chartjs.org/), and report abuse link (https://www.nuget.org/packages/Chart.js/2.9.3/ReportAbuse). The 'Dependencies' section shows 'No dependencies'.

NuGet Package Manager: ITW2020

Package source: [nuget.org](#)

Chart.js by Nick Downie, 835K downloads
Simple HTML5 Charts using the canvas element. v2.9.3

ChartJS by Developer Express Inc., 130K downloads
DevExtreme Web - Chart And Visualization - JavaScript Charting and Visualization library. v20.2.4

Chart.Mvc by Martino Bordin, 99,5K downloads
A .NET wrapper to generate charts using the popular Chart.js library (<http://www.chartjs.org/>). v1.0.0.2

ChartJSCore by mattosaurus, 75,3K downloads
A .NET Core library for generating Chart.js code. v1.5.8

chart.js.TypeScript.DefinitelyTyped by Jason Jarrett, 28,7K downloads
TypeScript Definitions (d.ts) for chart.js. Generated based off the DefinitelyTyped repository [git commit: 49e00c423a63b6e5b00cf0105c5fe6a2ac57336b]. <http://github.com/DefinitelyTyped> v0.1.1

ChartJs.Blazor by Marius Muntean, 92,6K downloads
ChartJs charts for Blazor. Compatible with client-side and server-side Blazor applications. v1.1.0

DlhSoft GanttChartHyperl library by DlhSoft, 76,7K downloads v5.3.17

Chart.js by Nick Downie, 835K downloads
Simple HTML5 Charts using the canvas element. v2.9.3

Installed: 2.9.3 Uninstall

Version: 2.9.3 Update

Options

Description
Simple HTML5 Charts using the canvas element.

Version: 2.9.3
Author(s): Nick Downie
License: [View License](#)
Date published: Thursday, November 28, 2019 (11/28/2019)
Project URL: <http://www.chartjs.org/>
Report Abuse: <https://www.nuget.org/packages/Chart.js/2.9.3/ReportAbuse>

Dependencies
No dependencies

