**INTRODUÇÃO AO DESENVOLVIMENTO DE APLICAÇÕES WEB**

**COM O GOOGLE MAPS API**



**SUMÁRIO**

[1. INTRODUÇÃO 4](#_Toc400886472)

[1.1 PÚBLICO ALVO 4](#_Toc400886473)

[1.2 PRÉ-REQUISITOS PARA O CURSO 4](#_Toc400886474)

[1.3 OBJETIVOS DO CURSO 4](#_Toc400886475)

[2. GOOGLE MAPS API 5](#_Toc400886476)

[2.1 O QUE É UMA API? 5](#_Toc400886477)

[2.2 API DO GOOGLE MAPS 5](#_Toc400886478)

[2.3 GUIA DO DESENVOLVEDOR 5](#_Toc400886479)

[3. GOOGLE MAPS NO MUNDO REAL 6](#_Toc400886480)

[3.1 APLICAÇÕES QUE UTILIZAM O GOOGLE MAPS 6](#_Toc400886481)

[3.2 DEMOS ONLINE 7](#_Toc400886482)

[4. GETTING STARTED 8](#_Toc400886483)

[4.1 GOOGLE MAPS API KEY 8](#_Toc400886484)

[4.2 PROJETO NO GITHUB 8](#_Toc400886485)

[5. DEMOS 9](#_Toc400886486)

[5.1 DEMO 0 (Beta) 9](#_Toc400886487)

[5.2 DEMO 1 – UM MAPA COM MAIS ESTILO 12](#_Toc400886488)

[5.3 DEMO 2 – MARKERS MANUAIS 13](#_Toc400886489)

[5.4 DEMO 3 – MARKERS DINÂMICOS 14](#_Toc400886490)

[5.5 DEMO 4 – TOO MANY MARKERS! 15](#_Toc400886491)

[5.6 DEMO 5 – UM MAPA PERSONALIZADO 16](#_Toc400886492)

[5.7 DEMO 6 – CARREGANDO PONTOS DE UM ARQUIVO .JSON 17](#_Toc400886493)

[5.8 DEMO 7 – CRIANDO ROTA ENTRE DOIS PONTOS 17](#_Toc400886494)

[6. LEVANDO SEU MAPA PARA O PRÓXIMO NÍVEL 18](#_Toc400886495)

[7. REFERÊNCIAS TÉCNICAS 19](#_Toc400886496)

[TIPOS DE MAPA: 19](#_Toc400886497)

# 1. INTRODUÇÃO

## 1.1 PÚBLICO ALVO

Desenvolvedores de software, entusiastas ou basicamente qualquer pessoa que queira aprender como desenvolver aplicações utilizando os recursos da API do Google Maps.

## 1.2 PRÉ-REQUISITOS PARA O CURSO

O intuito do curso é compartilhar conhecimentos e noções básicas sobre a API do Google Maps no entanto, alguns pré-requisitos são necessários para que todos possam aproveitar da mesma forma.

* Conhecimentos mínimos: HTML, CSS e Javascript (Básico)
* Conhecimentos desejáveis: HTML, CSS, Javascript (Intermediário) e jQuery

## OBJETIVOS DO CURSO

O curso tem como principais objetivos os seguintes itens em sua ordem cronológica:

* Entender os principais conceitos acerca da API de desenvolvimento do Google Maps
* Exemplificar utilizações do Google Maps em aplicações do mundo real
* Proporcionar um conhecimento inicial sobre como desenvolver aplicações Web que consumam os recursos do Google Maps (Prático)
* Sugestões de aprofundamento sobre o assunto: Levando sua aplicação Google Maps para o próximo nível.

# 2. GOOGLE MAPS API

## 2.1 O QUE É UMA API?

O termo API significa: *Application Programming Interface* (Interface de programação). Em resumo, uma API é uma especificação composta por um conjunto de recursos que podem ser consumidos por outros softwares.

## 2.2 API DO GOOGLE MAPS

A API do Google Maps descreve um conjunto de recursos que podem ser utilizados para adicionar mapas interativos em basicamente qualquer tipo de aplicação, embora seja mais frequente ver a sua utilização em aplicações Web/Mobile. A API do Google Maps é exposta através da linguagem Javascript portanto, para consumir seus recursos deve-se possuir um conhecimento básico nessa linguagem. A versão mais atual da API do Google Maps é a 3, também conhecida como V3. Essa será a versão adotada neste curso.

## 2.3 GUIA DO DESENVOLVEDOR

Um ponto muito importante é o guia de desenvolvimento da API do Google Maps. O guia na verdade é a documentação online da API do Google Maps. A documentação é o ponto mais importante e de mais valia para a vida dos desenvolvedores. É através dela que obtém-se informações de modo a conhecer e aprender a utilizar de forma correta todos os recursos que são oferecidos. A página chamada Guia do desenvolvedor pode ser acessada no seguinte endereço:  
<https://developers.google.com/maps/documentation/javascript/tutorial>

# 3. GOOGLE MAPS NO MUNDO REAL

Antes de começar com o desenvolvimento de aplicações utilizando os recursos da API do Google Maps, é interessante analisar cases de uso de aplicações do mundo real que incorporam o Google Maps. Dessa forma pode-se conhecer como as empresas estão utilizando o Google Maps, o que já existe no mercado e de uma forma mais visível, verificar o que pode-se criar com essa ferramenta.

## 3.1 APLICAÇÕES QUE UTILIZAM O GOOGLE MAPS

A utilização do Google Maps nas aplicações é atualmente bastante diverso. Não somente sendo utilizado para traçar rotas e indicar endereços, hoje o Google Maps é intensamente utilizado por diversas aplicações dos demais segmentos e das mais diversas maneiras. Alguns exemplos que fogem do tradicional podem ser:

* Rastreamento em tempo real
* Mapas de calor: Incidência de pessoas com vírus Ebola no mundo
* Mapas de indicadores: Mapas com indicadores utilizados pelos gestores das empresas.
  + Indicadores de quantas pessoas possuem smartphones Android no estado de São Paulo
  + Indicadores de status de servidores e links de internet

Como pode-se notar, a utilização do Google Maps em aplicações é quase que infinita. Desde uma simples plotagem de coordenadas à painéis de inteligência que fornecem feedback em tempo real para os gestores da organizações.

## 3.2 DEMOS ONLINE

Abaixo foram selecionados alguns exemplos reais de utilização do Google Maps

**Mapa com incidências de terremotos:**

<http://www.morethanamap.com/demos/visualization/earthquakes>

**Mapa de calor de população no mundo:**

<http://www.morethanamap.com/demos/visualization/population>

**Voos em tempo real:**

<http://www.morethanamap.com/demos/visualization/flights>

**Mapas com tweets durante uma viagem:**

<http://tweetedtrips.com/>

**Maps com incidência de assaltos e alertas de segurança/violência:**

<https://www.streetviolence.org/>

**Mais:**

<http://www.noupe.com/development/collection-of-the-coolest-uses-of-the-google-maps-api.html>

# 4. GETTING STARTED

## 4.1 GOOGLE MAPS API KEY

Antes de iniciar o desenvolvimento utilizando o Google Maps, é necessário obter uma chave da API. Essa chave é necessária para desenvolvedores que desejam utilizar o Google Maps em suas aplicações. Ela é útil pois posteriormente pode-se analisar dados referentes a utilização dos mapas em suas aplicações. Dados como quantidade de pessoas visualizando, o que fazem e etc.

**Nota**: A chave é gratuita e baste ter uma conta no Google para obtê-la. No entanto, só pode-se utilizar dela em aplicações distribuídas gratuitamente e existe um limite de utilização e trafego. Se sua aplicação é alguma aplicação privada e paga ou que vai ter muitos acessos, você deve contatar o serviço do Google voltado para empresas.

Para obter a Chave da API, basta seguir os passos no site da documentação:

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

## 4.2 PROJETO NO GITHUB

Todas as demos desse mini-curso estão disponíveis em um repositório no GitHub. Elas serão atualizadas e aprimoradas sempre. Também com o tempo serão adicionadas mais demos com novas funcionalidades e casos de uso, por isso confira sempre o repo de tempos em tempos! Recomenda-se clonar o repositório localmente para facilitar o andar do curso. O endereço é:

<https://github.com/joaopgrassi/Google-Maps-Mini-Curso>

# 5. DEMOS

## 5.1 DEMO 0 (Beta)

O nosso primeiro demo chamado de beta será nosso ponto de partida. Iremos adicionar um simples mapa em uma página HTML e explicar o funcionamento dos principais componentes necessários.

Para isso, siga as instruções:

1. Crie uma nova página HTML utilizando um editor HTML
2. Adicione a seguinte estrutura HTML:
3. <!DOCTYPE html**>**
4. **<html>**
5. **<head>**
6. **<meta** name="viewport" content="initial-scale=1.0, user-scalable=no" **/>**
8. <!-- Estilo da pagina -->
9. **<style** type="text/css"**>**
10. html { height: 100% }
11. body { height: 100%; margin: 0; padding: 0 }
12. #mapa { height:100%; }
13. **</style>**
15. <!-- Maps API Javascript -->
16. **<script** src="http://maps.googleapis.com/maps/api/js?sensor=false"**></script>**
18. <!-- Inicializa o mapa -->
19. **<script** type="text/javascript"**>**
21. function initialize() {
23. //Opcoes do mapa
24. var mapOptions = {
25. center: new google.maps.LatLng(-17, -50), //Aonde sera centralizado
26. zoom: 5, //nivel de zoom
27. mapTypeId: google.maps.MapTypeId.ROADMAP //Tipo de Mapa
28. };
30. var map = new google.maps.Map(document.getElementById("mapa"),
31. mapOptions);
32. }
33. **</script>**
35. **</head>**
36. **<body** onload="initialize()"**>**
37. <!-- Container do Mapa -->
38. **<div** id="mapa"**></div>**
39. **</body>**
40. **</html>**
41. Abra a página no navegador para visualizar o mapa.

Com o mapa funcionando vamos entender agora o que cada porção de código faz:

**Estilo CSS**

1. **<style** type="text/css"**>**
2. html { height: 100% }
3. body { height: 100%; margin: 0; padding: 0 }
4. #mapa { height:100%; }
5. **</style>**

Define o estilo css da página. Seta o HTML, o Body e o container do mapa (Div nomeada Mapa) com uma altura de 100%

**Carregando a API do Google Maps**

1. <!-- Maps API Javascript -->
2. **<script** src="http://maps.googleapis.com/maps/api/js?sensor=false"**></script>**

Insere na página o arquivo javascript da API do Google Maps. É nesse script que deve ser adicionado a sua chave da API.

**Corpo da página**

1. **<body** onload="initialize()"**>**
2. <!-- Container do Mapa -->
3. **<div** id="mapa"**></div>**
4. **</body>**

Essa é a porção da página que defini aonde nosso mapa será posicionado. Na tag <body> é registrado ao evento *onload* a função javascript *initialize().* Isso é, quando a página terminar de carregar, a função *initialize()*  será executada.

Abaixo temos a div “mapa”. Esse será nosso container do mapa. O Id da div é importante pois será usado posteriormente para a inicialização do mapa na página.

**Função *Initialize()***

1. <!-- Inicializa o mapa -->
2. **<script** type="text/javascript"**>**
4. function initialize() {
6. //Opcoes do mapa
7. var mapOptions = {
8. center: new google.maps.LatLng(-17, -50), //Aonde sera centralizado
9. zoom: 5, //nivel de zoom
10. mapTypeId: google.maps.MapTypeId.ROADMAP //Tipo de Mapa
11. };
13. var map = new google.maps.Map(document.getElementById("mapa"),
14. mapOptions);
15. }
16. **</script>**

Essa função é a responsável por criar o mapa em nossa página. Vamos entender o que ela faz:

* Para criar um mapa, precisamos criar uma nova instância da classe Map, que está contida na API que importamos em nossa página. A sintax para criar um mapa é:

*var mapa = new google.maps.Map( <container>, <opções>);*

***<container>****:* é o elemento HTML que será o container do mapa, ou seja, é aonde o mapa vai estar na página.

***<opções>:*** É um objeto JSON com as opções iniciais do mapa.

* Retornando ao nosso código acima, vemos então que:
  1. É criado um objeto ***mapOptions*** com as opções:
     1. Center: defini aonde o mapa vai centralizar ao abrir a página
     2. Zoom: Qual o nível de zoom do mapa
     3. [mapTypeId](#_TIPOS_DE_MAPA:): Qual o tipo de mapa que será utilizado
  2. Em seguida é criado o mapa passando a nossa div “mapa” como ***<container>*** e as opções criadas anteriormente como ***<opções>***

## 5.2 DEMO 1 – UM MAPA COM MAIS ESTILO

No primeiro demo vamos refatorar o código do nosso mapa beta. Iremos aplicar boas práticas de desenvolvimento, estilizar a página e incorporar jQuery no nosso desenvolvimento!



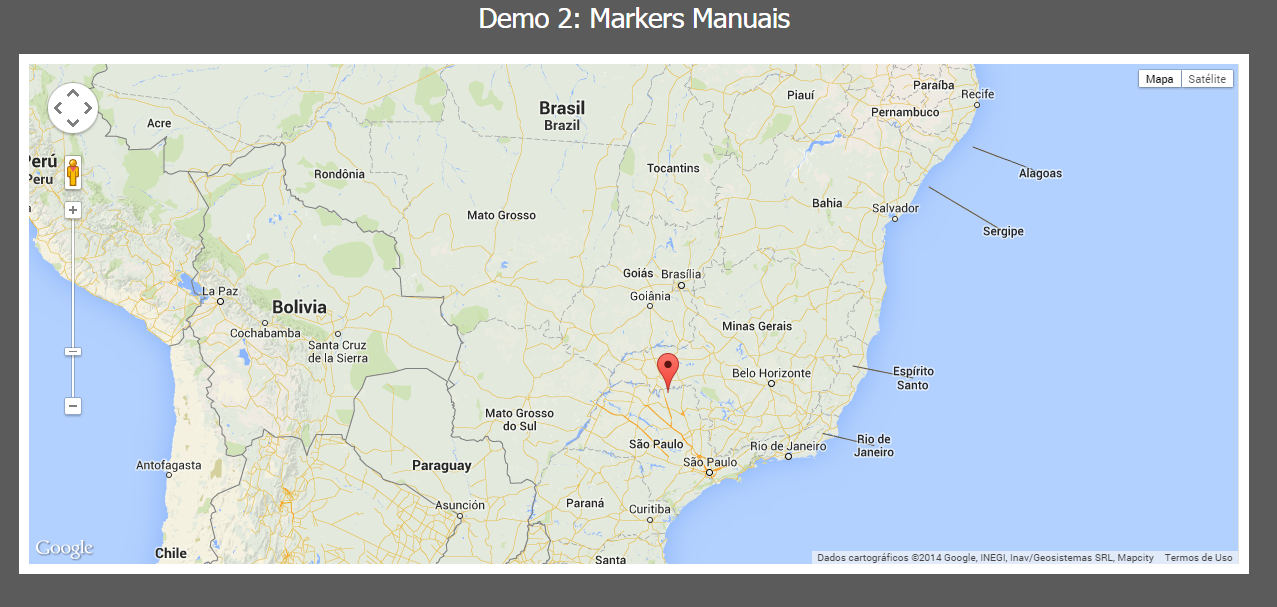
Demo 1: Preview do demo 1

## 5.3 DEMO 2 – MARKERS MANUAIS

No segundo demo, iremos iniciar a inserção de pontos (markers) em nosso mapa. Vamos entender os conceitos por trás deles, como são criados e como funciona a conexão dos markers com o mapa.

**Conceitos aprendidos:**

* Objeto LatLon da API do Google Maps
* Objeto Marker da API do Google Maps



Demo 2: Preview do demo 2

## 5.4 DEMO 3 – MARKERS DINÂMICOS

No demo 3 vamos criar uma estrutura em nossa página que irá expor a possibilidade de adicionarmos markers ao mapa através de coordenadas (Latitude/Longitude) e também ao clicar no mapa. Outra funcionalidade nesse demo será limpar os markers do mapa.

**Conceitos aprendidos:**

* Trabalhando com arrays de markers e variáveis globais
* Clicks em botões com jQuery
* Event Listeners



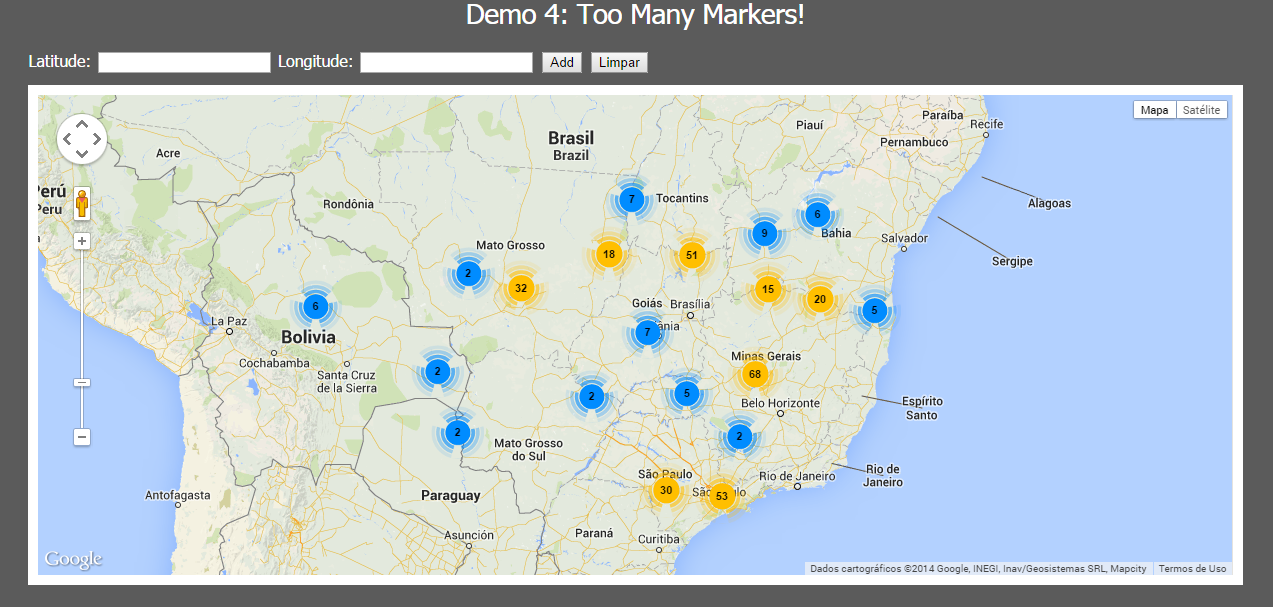
Demo 3: Preview do demo 3

## 5.5 DEMO 4 – TOO MANY MARKERS!

Depois de termos entendido como os markers funcionam, veremos como podemos melhorar a visualização do mapa agrupando os pontos. Para isso será introduzido um novo componente em nossa página, o Marker Clusterer.

**Conceitos aprendidos:**

* Bibliotecas auxiliares
* Agrupamento de pontos utilizando sistema de grid (markerclusterer)



Demo 4: Preview do demo 4

**Recursos deste demo:**

Docs:

<https://developers.google.com/maps/articles/toomanymarkers>

Repositório V3:

http://google-maps-utility-library-v3.googlecode.com/svn/trunk/markerclusterer/

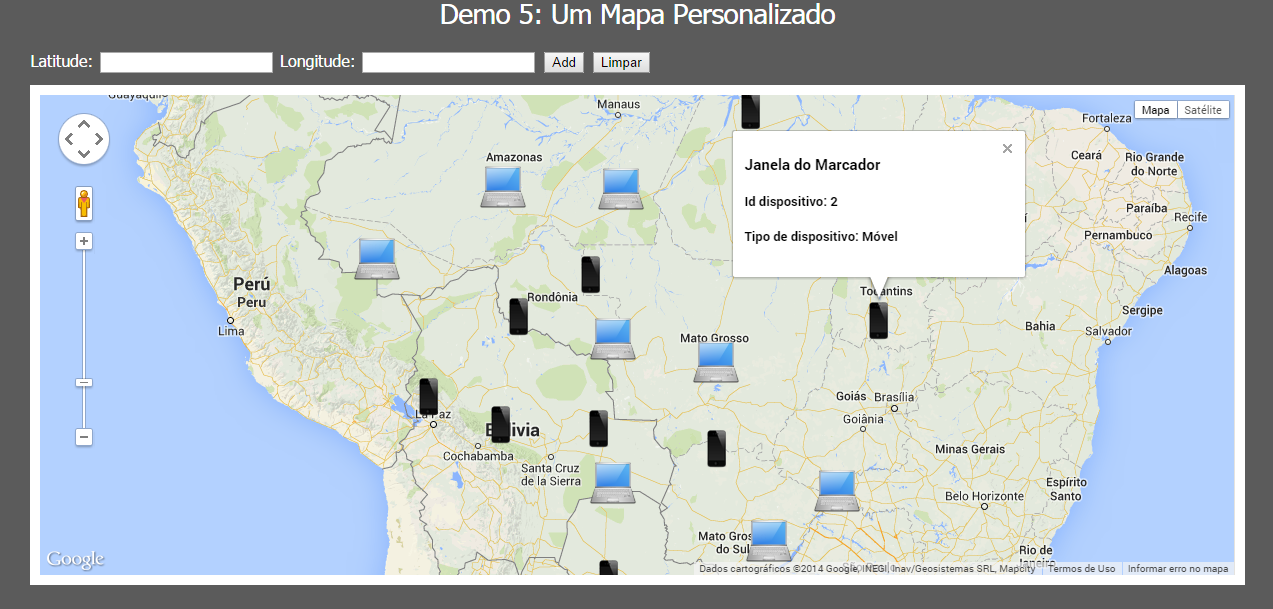
## 5.6 DEMO 5 – UM MAPA PERSONALIZADO

No demo 5 iremos ver algumas técnicas que podemos adotar para personalizar mais nosso mapa:

* Alterar o ícone padrão dos marcadores
* Alterar o ícone padrão dos agrupadores
* Criar janelas de informação

**Conceitos aprendidos:**

* Sobrescrevendo ícones padrões
* Janela do marker (InfoWindow)
* Estilizando a infoWindow
* Manipulação de template com jQuery



Demo 5: Preview do demo 5

## 5.7 DEMO 6 – CARREGANDO PONTOS DE UM ARQUIVO .JSON

Neste demo iremos consumir pontos de um arquivo .json e adicioná-los ao mapa. Essa técnica é muito utilizada e também é similar a casos reais onde realizamos chamadas ajax para algum serviço ou api para recuperar os pontos.

## 5.8 DEMO 7 – CRIANDO ROTA ENTRE DOIS PONTOS

No último demo vamos aprender como criar um mapa que exibe rotas entre dois endereços, similar ao sistema de rotas do próprio Google Maps

# 6. LEVANDO SEU MAPA PARA O PRÓXIMO NÍVEL

1. Vasto conteúdo de tutoriais, exemplos e casos de uso: <http://www.geocodezip.com/>
2. Solução criada para mapas do tipo ROADMAP. Markers na exata mesma posição não são visualizados: <https://github.com/jawj/OverlappingMarkerSpiderfier>
3. Aplicar auto zoom no local aonde os markers estão: <http://blog.shamess.info/2009/09/29/zoom-to-fit-all-markers-on-google-maps-api-v3/>

# 7. REFERÊNCIAS TÉCNICAS

## TIPOS DE MAPA:

A API do Google Maps oferece diversos tipos de mapa. O mais utilizado é o tipo ROADMAP. No entanto, existem outros tipos de mapas que dependendo do cenário podem ser utilizados:

ROADMAP



SATELLITE

****

HYBRID

****

TERRAIN

****