

# **SISTEM INFORMASI GEOGRAFIS**



**OLEH :**

Kadek Arie Anggara Putra

2105541092

**UNIVERSITAS UDAYANA  
FAKULTAS TEKNIK  
PROGRAM STUDI TEKNIK ELEKTRO  
2024**

## Tugas Google Maps

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4 <title>WebGIS Google Maps Tugas 1</title>
5 <meta name="viewport" content="initial-scale=1.0, user-scalable=no" />
6 <meta charset="utf-8">
7 <style type="text/css">
8   html { height: 100% }
9   body { height: 100%; margin: 0; padding: 0 }
10  #map-canvas { width: 100%; height: 700px; }
11 </style>
12 <script type="text/javascript" src="https://maps.google.com/maps/api/js?key=AIzaSyAs2Tvuok7nMf09wjK3owAlzp-ai9wzby"></script>
13
14 <script>
15   function initialize() {
16     var mapOptions = {
17       center: new google.maps.LatLng(-8.7467357, 115.1668024),
18       zoom: 13,
19       mapTypeId: google.maps.MapTypeId.ROADMAP
20     };
21
22     var map = new google.maps.Map(document.getElementById("map-canvas"), mapOptions);
23
24     var marker = new google.maps.Marker({
25       position: mapOptions.center,
26       map: map,
27       draggable: true,
28       title: 'Marker'
29     });
30
31     var infowindow = new google.maps.InfoWindow();
32
33     google.maps.event.addListener(marker, 'dragend', function () {
34       var newPosition = marker.getPosition();
35       infowindow.setContent('Koordinat: ' + newPosition.lat().toFixed(7) + ', ' + newPosition.lng().toFixed(7));
36       infowindow.open(map, marker);
37     });
38
39     google.maps.event.addListener(marker, 'click', function () {
40       infowindow.setContent(marker.title);
41       infowindow.open(map, marker);
42     });
43   }
44   google.maps.event.addDomListener(window, 'load', initialize);
45 </script>
46 </head>
47 <body>
48   <h1>Objek Peta Google Maps dengan Draggable Marker</h1>
49   <div id="map-canvas"></div>
50 </body>
51 </html>
```

**Gambar 1.** Tampilan Keseluruhan *Code*

Google Maps API adalah sebuah layanan (service) yang diberikan oleh Google kepada para pengguna untuk memanfaatkan Google Map dalam mengembangkan aplikasi. Google Maps API menyediakan beberapa fitur untuk memanipulasi peta, dan menambah konten melalui berbagai jenis services yang dimiliki, serta mengizinkan kepada pengguna untuk membangun aplikasi enterprise di dalam websitenya.

## Penjelasan *Code*

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4 <title>WebGIS Google Maps Tugas 1</title>
5 <meta name="viewport" content="initial-scale=1.0, user-scalable=no" />
6 <meta charset="utf-8">
7 <style type="text/css">
8   html { height: 100% }
9   body { height: 100%; margin: 0; padding: 0 }
10  #map-canvas { width: 100%; height: 700px; }
11 </style>
12 <script type="text/javascript" src="https://maps.google.com/maps/api/js?key=AIzaSyAs2Tvuok7nMf09wjK3owAlzp-ai9wzby"></script>
13
```

**Gambar 2.** *Code line* 1 sampai 13

Pada screenshot *code* diatas menampilkan `<!DOCTYPE html>` dimana menyatakan bahwa dokumen ini adalah dokumen HTML5, lalu `<html>` sendiri merupakan elemen akar dari dokumen HTML. Lalu pada bagian Bagian `<head>` berisi metadata dokumen, gaya CSS, dan skrip JavaScript. `<title>`: menentukan judul dokumen yang muncul di tab browser, `<meta name="viewport" content="initial-scale=1.0, user-scalable=no" />`: mengatur tampilan dan perilaku halaman pada perangkat seluler lalu `initial-scale=1.0` mengatur skala awal halaman dan `user-scalable=no` mencegah pengguna untuk mengubah skala halaman, `<meta charset="utf-8">`: menentukan karakter encoding dokumen sebagai UTF-8. Lalu untuk `<style>`: mendefinisikan gaya CSS untuk elemen HTML, dimana `html { height: 100% }`: mengatur tinggi elemen html sebesar 100%, lalu `body { height: 100%; margin: 0; padding: 0 }`: mengatur tinggi elemen body sebesar 100%, dan menghilangkan margin dan padding dan `#map-canvas { width: 100%; height: 700px; }`: mengatur elemen dengan id map-canvas memiliki lebar 100% dan tinggi 700px. Terakhir `<script>`: memuat pustaka Google Maps API menggunakan kunci API yang disediakan yakni `AIzaSyAs2TvuOK7nMHfo9wjK3owAlzp-ai9wzbY`.

```
14 <script>
15     function initialize() {
16         var mapOptions = {
17             center: new google.maps.LatLng(-8.7467357, 115.1668024),
18             zoom: 13,
19             mapTypeId: google.maps.MapTypeId.ROADMAP
20         };
21
22         var map = new google.maps.Map(document.getElementById("map-canvas"), mapOptions);
23
24         var marker = new google.maps.Marker({
25             position: mapOptions.center,
26             map: map,
27             draggable: true,
28             title: 'Marker'
29         });
30
31         var infowindow = new google.maps.InfoWindow();
32     }
```

**Gambar 3.** Code line 14 sampai 32

Pada screenshot *code* diatas menampilkan skrip dari JavaScript, dengan fungsi `initialize` berisi semua logika untuk menginisialisasi dan mengatur peta Google Maps serta marker di dalamnya. Fungsi ini akan dipanggil saat halaman selesai dimuat. Lalu terdapat objek `mapOptions` digunakan untuk menyimpan opsi konfigurasi untuk peta Google Maps. Dimana `center`: mengatur pusat peta pada

koordinat tertentu menggunakan `google.maps.LatLng` dengan lintang `-8.7467357` dan bujur `115.1668024`, `zoom`: mengatur tingkat zoom peta pada level 13, `mapTypeId`: menentukan tipe peta yang akan ditampilkan dan `google.maps.MapTypeId.ROADMAP` menunjukkan bahwa tipe peta yang digunakan adalah peta jalan. Lalu ada objek `map`, `map` adalah objek peta Google Maps yang dibuat dengan memanggil konstruktor, lalu `google.maps.Map` adalah Elemen HTML dengan id `map-canvas` digunakan sebagai kontainer peta, dan `mapOptions` digunakan sebagai konfigurasi peta. Lalu objek `marker`, yang dibuat dengan memanggil konstruktor `google.maps.Marker`. `Marker` ini ditempatkan di peta dengan pengaturan; `position`: mengatur posisi `marker` sesuai dengan `mapOptions.center`. Lalu `map`: menetapkan peta yang akan menampilkan `marker` (`map`). Selanjutnya `draggable`: Mengatur `marker` menjadi `draggable` (dapat dipindahkan dengan cara di-*drag*). Terakhir pada objek `marker` yakni `title`: menetapkan judul `marker` yang akan ditampilkan sebagai tooltip. Objek terakhir yakni `infoWindow`. `google.maps.InfoWindow`, `Info Window` ini nantinya bisa digunakan untuk menampilkan informasi tambahan saat `marker` di-klik atau di-*drag*

```
33 google.maps.event.addListener(marker, 'dragend', function () {
34     var newPosition = marker.getPosition();
35     infoWindow.setContent('Koordinat: ' + newPosition.lat().toFixed(7) + ', ' + newPosition.lng().toFixed(7));
36     infoWindow.open(map, marker);
37 });
38
39 google.maps.event.addListener(marker, 'click', function () {
40     infoWindow.setContent(marker.title);
41     infoWindow.open(map, marker);
42 });
43
44 }
45 google.maps.event.addDomListener(window, 'load', initialize);
46 </script>
47 </head>
48 <body>
49     <h1>Objek Peta Google Maps dengan Draggable Marker</h1>
50     <div id="map-canvas"></div>
51 </body>
52 </html>
```

**Gambar 4.** Code line 33 sampai 52

Pada screenshot *code* diatas menampilkan `google.maps.event.addListener(marker, 'dragend', function () { ... });` fungsi `addListener` menambahkan event listener ke objek `marker`. Lalu event yang didaftarkan adalah `'dragend'`, yang akan dipicu saat pengguna selesai memindahkan atau men-*drag* `marker`. Lalu `var newPosition = marker.getPosition();` mendapatkan posisi baru dari `marker` setelah dipindahkan. `getPosition()` mengembalikan objek `LatLng` yang berisi koordinat `marker` saat ini.

Selanjutnya `infoWindow.setContent('Koordinat: ' + newPosition.lat().toFixed(7) + ', ' + newPosition.lng().toFixed(7));` mengatur konten `infoWindow` dengan string yang berisi koordinat baru marker. `newPosition.lat().toFixed(7)` dan `newPosition.lng().toFixed(7)` digunakan untuk mendapatkan lintang dan bujur dengan 7 digit setelah titik desimal. Setelah itu `infoWindow.open(map, marker);` membuka `infoWindow` pada peta atau map dengan posisi di atas marker (`marker`). Selanjutnya pada *code* diatas juga menampilkan `google.maps.event.addListener(marker, 'click', function () { ... });` fungsi `addListener` menambahkan event listener ke objek marker. Lalu event yang didaftarkan adalah 'click', yang akan dipicu saat pengguna mengklik marker. Lalu terdapat `infoWindow.setContent(marker.title);` untuk mengatur konten `infoWindow` dengan teks yang disimpan dalam properti `title` dari marker dan `infoWindow.open(map, marker);` untuk membuka `infoWindow` pada peta (`map`) dengan posisi di atas marker (`marker`). Terdapat pula `google.maps.event.addDomListener(window, 'load', initialize);` fungsi `addDomListener` menambahkan event listener ke objek DOM (`window`). Lalu event yang didaftarkan adalah 'load', yang akan dipicu saat halaman web selesai dimuat. Lalu fungsi yang dipanggil saat event 'load' terjadi adalah `initialize`, yang bertugas menginisialisasi peta dan marker. Terakhir bagian `<body>` berisi konten yang akan ditampilkan di halaman web yakni `<h1>`: menampilkan judul utama halaman dan `<div id="map-canvas"></div>`: elemen `div` dengan id `map-canvas`, yang akan digunakan sebagai kontainer untuk peta Google Maps.

## Hasil Akhir Peta

