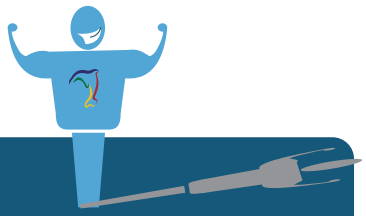
autentia

Avenida de Castilla,1 - Edificio Best Point - Oficina 21B
28830 San Fernando de Henares (Madrid)
tel./fax: +34 91 675 33 06
info@autentia.com - **www.autentia.com**

¿**Qué ofrece** Autentia Real Business Solutions S.L?

Somos su empresa de **Soporte a Desarrollo Informático**.
Ese apoyo que siempre quiso tener...

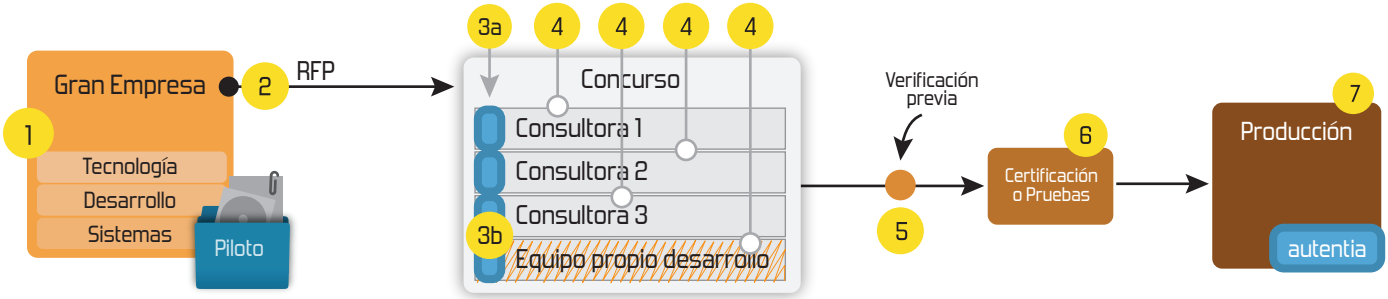
1. Desarrollo de componentes y proyectos a medida



2. Auditoría de código y recomendaciones de mejora

3. Arranque de proyectos basados en nuevas tecnologías

1. Definición de frameworks corporativos.
2. Transferencia de conocimiento de nuevas arquitecturas.
3. Soporte al arranque de proyectos.
4. Auditoría preventiva periódica de calidad.
5. Revisión previa a la certificación de proyectos.
6. Extensión de capacidad de equipos de calidad.
7. Identificación de problemas en producción.

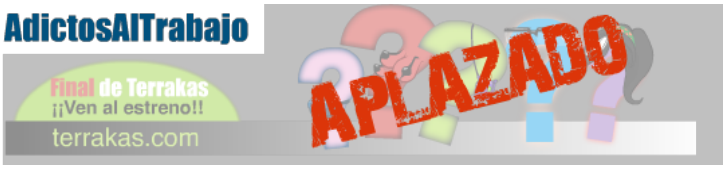


4. Cursos de formación (impartidos por desarrolladores en activo)

- Spring MVC, JSF-PrimeFaces /RichFaces, HTML5, CSS3, JavaScript-jQuery
- Gestor portales (Liferay)
- Gestor de contenidos (Alfresco)
- Aplicaciones híbridas
- Control de autenticación y acceso (Spring Security)
- UDDI
- Web Services
- Rest Services
- Social SSO
- SSO (Cas)
- JPA-Hibernate, MyBatis
- Motor de búsqueda empresarial (Solr)
- ETL (Talend)
- Dirección de Proyectos Informáticos.
- Metodologías ágiles
- Patrones de diseño
- TDD
- Tareas programadas (Quartz)
- Gestor documental (Alfresco)
- Inversión de control (Spring)
- BPM (jBPM o Bonita)
- Generación de informes (JasperReport)
- ESB (Open ESB)

Compartimos nuestro conocimiento en:
www.adictosaltrabajo.com

Para más información visítenos en:
www.autentia.com

AdictosAlTrabajo


autentia
Soporte a desarrollo informático
Hosting patrocinado por enredados

Entra en Adictos a través de
E-mail
Contraseña
Entrar Deseo registrarme
Olvidé mi

Inicio Quiénes somos Formación Comparador de salarios Nuestros libros Más

» Estás en: Inicio Tutoriales Clustering de puntos en Google Maps v3



Rubén Aguilera Díaz-Heredero

Consultor tecnológico de desarrollo de proyectos informáticos.

Ingeniero en Informática, especialidad en Ingeniería del Software

Puedes encontrarme en Autentia: Ofrecemos servicios de soporte a desarrollo, factoría y formación

Somos expertos en Java/J2EE

Ver todos los tutoriales del autor

Fecha de publicación del tutorial: 2013-05-16 Tutorial visitado 9 veces Descargar en PDF

Clustering de puntos en Google Maps v3

0. Índice de contenidos.

- 1. Entorno
- 2. Introducción
- 3. Vamos al lío
- 4. Referencias

1. Entorno

Este tutorial está escrito usando el siguiente entorno:

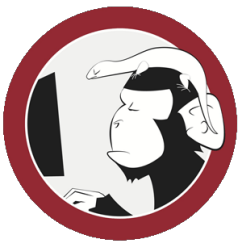
- Hardware: Portátil Mac Book Pro 17" (2,6 Ghz Intel Core i7, 8 GB DDR3)
- Sistema Operativo: Mac OS X Snow Leopard 10.6.4
- API Google Maps
- google-maps-utility-library-v3

2. Introducción

Cuando trabajamos en aplicaciones de geoposicionamiento con Google Maps nos podemos encontrar con que en base al zoom seleccionado y al número de puntos que haya que visualizar, éstos salgan muy juntos y no se puedan distinguir uno de otros como en el ejemplo que se muestra a continuación:

Catálogo de servicios Autentia





Síguenos a través de:

Últimas Noticias

» Atención, APLAZADO Estreno último capítulo de Terrakas

» Vendedor: Soy inseguro, filtra o elige por mi: si quieres que te compre.

» Comentando el libro: El arte de pensar, de Rolf Dobelli

» Ya está a la venta mi segundo libro: Planifica tu éxito, de aprendiz a empresario

» Ya esta disponible en eBook mi primer libro: Informática Profesional

Histórico de noticias



La solución conceptual para este problema es agrupar los puntos para mostrar uno solo, que pinchando sobre él vaya mostrando el resto. Este es el concepto de clustering y la forma más sencilla de implementarlo en Google Maps es utilizar las utilidades que están implementadas en la librería Marker Clusterer del proyecto google-maps-utility-library-v3 que se distribuye bajo licencia Apache 2.0, con lo que llegado el caso se podría utilizar en proyectos con fines comerciales.

3. Vamos al lío

En primer lugar vamos a utilizar el API de Google Maps para visualizar el mapa que hemos visto en la introducción. Para ello vamos a crear un HTML con el siguiente contenido:

view plain print ?

```
<!DOCTYPE>
<html>
  <head>
    <meta http-equiv="content-type" content="text/html; charset=utf-8" />
    <title>Ejemplo sin Cluster
    <style type="text/css">
      body {
        margin: 0;
        padding: 10px 20px 20px;
        font-family: Arial;
        font-size: 16px;
      }

      #map-container {
        padding: 6px;
        border-width: 1px;
        border-style: solid;
        border-color: #ccc #ccc #999 #ccc;
        -webkit-box-shadow: rgba(64, 64, 64, 0.5) 0 2px 5px;
        -moz-box-shadow: rgba(64, 64, 64, 0.5) 0 2px 5px;
        box-shadow: rgba(64, 64, 64, 0.1) 0 2px 5px;
        width: 600px;
      }

      #map {
        width: 600px;
        height: 400px;
      }

    </style>
    <script src="http://maps.google.com/maps/api/js?sensor=false">
    <script type="text/javascript" src="data.json">
    <script type="text/javascript">
      function initialize() {
        var center = new google.maps.LatLng(50, 20);

        var map = new google.maps.Map(document.getElementById('map'), {
          zoom: 3,
          center: center,
          mapTypeId: google.maps.MapTypeId.ROADMAP
        });

        for (var i = 0; i < 100; i++) {
          var datadataPhoto = data.photos[i];
          var latLng = new google.maps.LatLng(dataPhoto.latitude,
              dataPhoto.longitude);
          var marker = new google.maps.Marker({
            position: latLng,
            map: map,
          });
        }
      }
      google.maps.event.addDomListener(window, 'load', initialize);
    </script>
  </head>
  <body>
    <h3>Ejemplo sin Cluster
    <div id="map-container"><div id="map">
```

</body> </html>

En este ejemplo utilizamos el API de Google Maps para, a partir de los datos almacenados en el fichero data.json, ir posicionando los puntos en el mapa. El fichero data.json tendría este aspecto:

Últimos Tutoriales

» Xcode Guía Básica Test Unitarios

» Haciendo BDD con Cucumber

» Manejo de test con TestLink

» Prototipado de pantallas con Pencil

» Como testear aplicaciones en Ember.js

Últimos Tutoriales del Autor

» Resolver problema LockTimeoutException en Spring Web Flow

» Trabajar con tablas en JasperReport

» Aspectos avanzados de Sitemesh

» Introducción a Sitemesh

» Robolectric: aplicando TDD en Android

Últimas ofertas de empleo

2011-09-08
Comercial - Ventas - MADRID.

2011-09-03
Comercial - Ventas - VALENCIA.

2011-08-19
Comercial - Compras - ALICANTE.

2011-07-12
Otras Sin catalogar - MADRID.

2011-07-06
Otras Sin catalogar - LUGO.

```
view plain  print  ?
01. var data = { "count": 10785236,
02.  "photos": [{"photo id": 27932, "photo title": "Atardecer en Embalse", "photo url": "http://www.panoramio.com/photo/27932", "photo file u
    panoramio/photos/medium/27932.jpg", "longitude": -64.404945, "latitude": -32.202924, "width": 500, "height": 375, "upload_date": "25 June
03. ,
04. {"photo id": 522084, "photo title": "In Memoriam Antoine de Saint Exupéry", "photo url": "http://www.panoramio.com/photo/522084", "photo
    panoramio/photos/medium/522084.jpg", "longitude": 17.470493, "latitude": 47.867077, "width": 500, "height": 350, "upload_date": "21 Janua
05. ,
06. {"photo id": 1578881, "photo title": "Rosina Lamberti,Sunset,Templestowe , Victoria, Australia", "photo url": "http://www.panoramio.com/p
    panoramio/photos/medium/1578881.jpg", "longitude": 145.141754, "latitude": -37.766372, "width": 500, "height": 474, "upload_date": "01 Ap
07. ,
08. {"photo id": 97671, "photo title": "kin-dza-
    dza", "photo url": "http://www.panoramio.com/photo/97671", "photo file url": "http://mw2.google.com/mw-
    panoramio/photos/medium/97671.jpg", "longitude": 30.785408, "latitude": 46.639301, "width": 500, "height": 375, "upload_date": "09 Decemb
09. ,
10. {"photo id": 25514, "photo title": "Arenal", "photo url": "http://www.panoramio.com/photo/25514", "photo file url": "http://mw2.google.co
    panoramio/photos/medium/25514.jpg", "longitude": -84.693432, "latitude": 10.479372, "width": 500, "height": 375, "upload_date": "17 June
11. ]}
```

Estos son solo algunos de los puntos. Manteniendo esta estructura se pueden añadir los que queramos.

Si ahora visualizamos este fichero desde cualquier navegador web veríamos algo muy parecido a lo que se mostraba en la introducción.

Vamos a ver lo sencillo que es hacer clustering con la librería antes mencionada. Simplemente añadimos la librería y en vez de ir introduciendo uno por uno los puntos en el mapa, los almacenamos en un array que utilizaremos como parámetro de entrada para la construcción del clustering.

Este sería el fichero HTML con cluster:

```
view plain  print  ?
01. <meta http-equiv="content-type" content="text/html; charset=utf-8">
02. <title>MarkerClusterer v3 Example</title>
03.
04. <style type="text/css">
05.   body {
06.     margin: 0;
07.     padding: 10px 20px 20px;
08.     font-family: Arial;
09.     font-size: 16px;
10.   }
11.
12.   #map-container {
13.     padding: 6px;
14.     border-width: 1px;
15.     border-style: solid;
16.     border-color: #ccc #ccc #999 #ccc;
17.     -webkit-box-shadow: rgba(64, 64, 64, 0.5) 0 2px 5px;
18.     -moz-box-shadow: rgba(64, 64, 64, 0.5) 0 2px 5px;
19.     box-shadow: rgba(64, 64, 64, 0.1) 0 2px 5px;
20.     width: 600px;
21.   }
22.
23.   #map {
24.     width: 600px;
25.     height: 400px;
26.   }
27.
28. </style>
29.
30. <script src="http://maps.google.com/maps/api/js?sensor=false"></script><script src="http://maps.gstatic.com/intl/es_ALL/mapfiles/api-
    3/13/0/main.js" type="text/javascript"></script>
31. <script type="text/javascript" src="data.json"></script>
32. <script type="text/javascript" src="markerclusterer.js"></script>
33.
34. <script type="text/javascript">
35.   function initialize() {
36.     var center = new google.maps.LatLng(50, 20);
37.
38.     var map = new google.maps.Map(document.getElementById('map'), {
39.       zoom: 3,
40.       center: center,
41.       mapTypeId: google.maps.MapTypeId.ROADMAP
42.     });
43.
44.     var markers = [];
45.     for (var i = 0; i < 100; i++) {
46.       var datadataPhoto = data.photos[i];
47.       var latLng = new google.maps.LatLng(dataPhoto.latitude,
48.           dataPhoto.longitude);
49.       var marker = new google.maps.Marker({
50.         position: latLng
51.       });
52.       var marker = new google.maps.Marker({
53.             position: latLng,
54.             map: map,
55.         });
56.       markers.push(marker);
57.     }
58.     var markerCluster = new MarkerClusterer(map, markers);
59.   }
60.   google.maps.event.addDomListener(window, 'load', initialize);
61. </script>
62.
63.
64. <h3>Ejemplo con clustering</h3>
65. <div id="map-container"><div id="map"><div style="overflow: hidden;"></div></div></div>
```

El resultado debería ser algo parecido a esto:



4. Referencias

Estas son las referencias utilizadas para este tutorial:

- Google Maps Utility Library V3
- Marker Clusterer

Cualquier duda o sugerencia en la zona de comentarios.

Saludos.

A continuación puedes evaluarlo:

Regístrate para evaluarlo

Por favor, vota +1 o compártelo si te pareció interesante

Share | 0

Anímate y coméntanos lo que pienses sobre este **TUTORIAL:**

» Registrate y accede a esta y otras ventajas **«**

SOME RIGHTS RESERVED Esta obra está licenciada bajo licencia Creative Commons de Reconocimiento-No comercial-Sin obras derivadas 2.5

IMPULSA Impulsores Comunidad ¿Ayuda?

0 personas han traído clicks a esta página

sin clicks