

API SITNA, API Javascript para visualización de datos georreferenciados en aplicaciones web

F. Lacunza Prieto⁽¹⁾

⁽¹⁾ Área Geoweb, Departamento de Sistemas de Información Territorial, Tracasa, flacunza@tracasa.es.

RESUMEN

La API SITNA es una API JavaScript creada para facilitar la representación de información georreferenciada sobre visores de mapas en las aplicaciones web del Gobierno de Navarra.

Se ha intentado que la API sea sencilla de usar para un desarrollador sin grandes conocimientos GIS.

Por un lado, aunque se apoya en diversas bibliotecas JavaScript de terceros, está completamente autocontenida, cargando dinámicamente recursos y liberando al desarrollador de la responsabilidad de cargar en la página nada más que el script de la API.

Por otro, su configuración por defecto está definida para que la creación de una instancia de mapa sin ningún tipo de parametrización ni configuración ofrezca una visualización de un mapa de Navarra maquetoado con varias herramientas de uso común disponibles al usuario.

La API contiene un conjunto de operaciones habituales de los visores de mapas, siguiendo las necesidades específicas de los desarrollos en el Gobierno de Navarra. Por ejemplo, se pueden representar datos de servicios geográficos WMS y WMTS y se realizan búsquedas mediante WFS. También se puede integrar contenido de ficheros KML y superponer marcadores en puntos sobre el mapa.

La API está basada en OpenLayers 2 ó 3; la versión concreta se elige por configuración. Esta actúa como un envoltorio que traduce la jerarquía de objetos de OpenLayers a una propia. De este modo para el desarrollador es transparente el radical cambio que ha sufrido OpenLayers con el paso de la versión 2 a la 3.

El aspecto y comportamiento de los visores creados con la API SITNA puede ser personalizado con un sistema de carga dinámica de maquetaciones. La maquetación por defecto se adapta dinámicamente a dispositivos móviles.

Paralelamente se ha creado con la API una página enlazable desde un IFRAME para poder visualizar archivos KML en páginas HTML sin ningún tipo de desarrollo.

Palabras clave: Jornadas, SIG, software libre, Girona.

INTRODUCCIÓN

En los últimos años los equipos de desarrollo de aplicaciones web para el Gobierno de Navarra se han encontrado con la necesidad de incrustar mapas interactivos en sus productos, aprovechando los servicios OGC [1] que ofrece la Infraestructura de Datos Espaciales de Navarra (IDENA) [2] y los datos geográficos que ofrece la administración dentro de la iniciativa OpenData [3].

Existen varias bibliotecas de desarrollo JavaScript establecidas que permiten cubrir esa necesidad, como la API de Google Maps [4], OpenLayers [5][6] o Leaflet [7], pero o tienen una curva de aprendizaje dura para un desarrollador sin experiencia en GIS o no facilitan el uso del sistema de referencia espacial ETRS89 / UTM zone 30N, sistema por defecto en los productos geográficos de IDENA.

Por todo ello, a finales de 2013 se empezó a desarrollar una API JavaScript que permitiera incrustar visores de mapas en páginas web y que suavizara los escollos mencionados anteriormente, cumpliendo dos requisitos: máxima facilidad de uso para un neófito en GIS y capacidad de consumir los distintos servicios OGC de IDENA sin configuración adicional.

CARACTERÍSTICAS GENERALES

La API SITNA [8] actualmente puede consumir servicios WMS y WMTS, y visualizar documentos KML. Además consume servicios WFS para realizar búsquedas por texto de entidades geográficas.

En la jerarquía de objetos el objeto principal es de la clase `SITNA.Map`, que representa un mapa incrustado en un elemento HTML de una página. Métodos del objeto permiten añadir capas y marcadores de puntos al mapa y controles de usuario al visor. Cada control tiene asociada una plantilla HTML, de forma que es fácil personalizar su aspecto.

Otro objeto importante es `SITNA.Cfg`, que almacena la configuración general. Esta configuración puede ser sobrescrita añadiendo parámetros a los constructores de objetos de la API, pero tiene unos valores por defecto tales que la instanciación de un objeto de la clase `SITNA.Map` sin parámetros confecciona un mapa interactivo centrado en Navarra con el sistema de referencia espacial ETRS89 / UTM zone 30N, una serie de mapas de fondo disponibles y varios controles de usuario con herramientas de utilidad (búsquedas, mediciones, etc.), todo ello con una maquetación acorde con el estilo corporativo del Gobierno de Navarra. Como ejemplo, la siguiente página HTML es un visor como el que se puede ver en la figura 1.

```
<!DOCTYPE html>
<html>
  <body>
    <div id="map"></div>
    <script src="//sitna.tracasa.es/api/"></script>
    <script>
      var map = new SITNA.Map('map');
    </script>
  </body>
</html>
```



Figura 1: Aspecto por defecto de un mapa creado con la API SITNA.

Con sencillos cambios en la configuración se puede crear un visor centrado en cualquier otra región geográfica y en cualquier sistema de referencia espacial.

BÚSQUEDAS

Un control de usuario de especial interés que ofrece la API SITNA es el cuadro de búsquedas, que tiene funcionalidad de autocompletado de texto y que se apoya en servicios WFS para funcionar. Actualmente el control soporta la búsqueda de coordenadas y de municipios de Navarra, y ya se ha desarrollado la búsqueda de direcciones postales y referencias catastrales. Se puede observar el funcionamiento de este control en la figura 2.

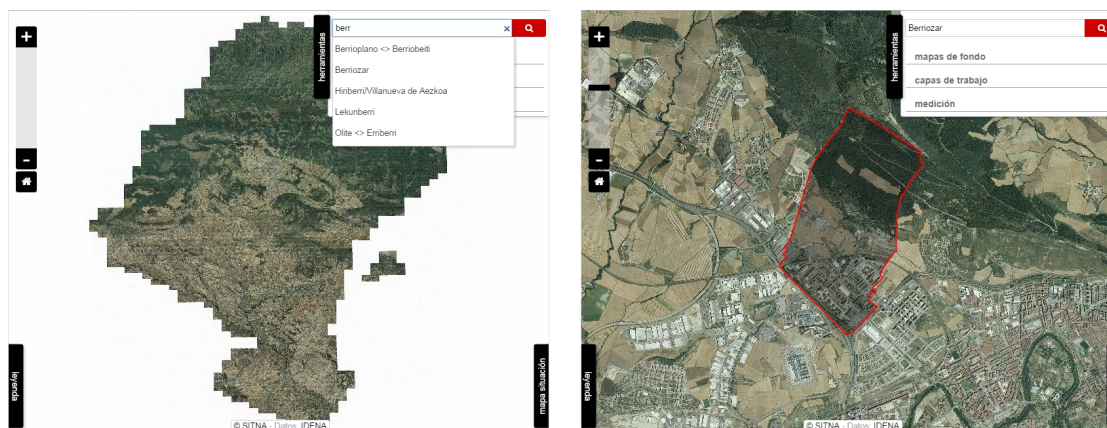


Figura 2: Búsqueda de un municipio.

MARCADORES

Actualmente el uso principal de la API SITNA es mostrar en un mapa puntos de interés. Para ello esta ofrece la capacidad de añadir marcadores en coordenadas del mapa, con aspecto configurable y con información asociada. A dicha información se accede mediante un bocado desplegable que se muestra al pulsar sobre el marcador.

Cada marcador puede asignarse a un grupo. Los grupos de marcadores aparecen en la tabla de contenidos y en la leyenda.

A continuación se muestra un fragmento de código con un ejemplo de creación de marcador con datos asociados. El resultado es el de la figura 3.

```
map.addMarker([-1.605691, 42.060453], { /* Coordenada */  
  cssClass: "kiosko", /* clase CSS que determina el icono  
  utilizado */  
  width: 40,  
  height: 40,  
  data: { /* datos asociados */  
    "Nombre": "Plaza de la Constitución, Tudela",  
    "Sitio web": "http://www.tudela.es/"  
  },  
  showPopup: true /* el marcador se dibuja con su información  
  desplegada */  
});
```

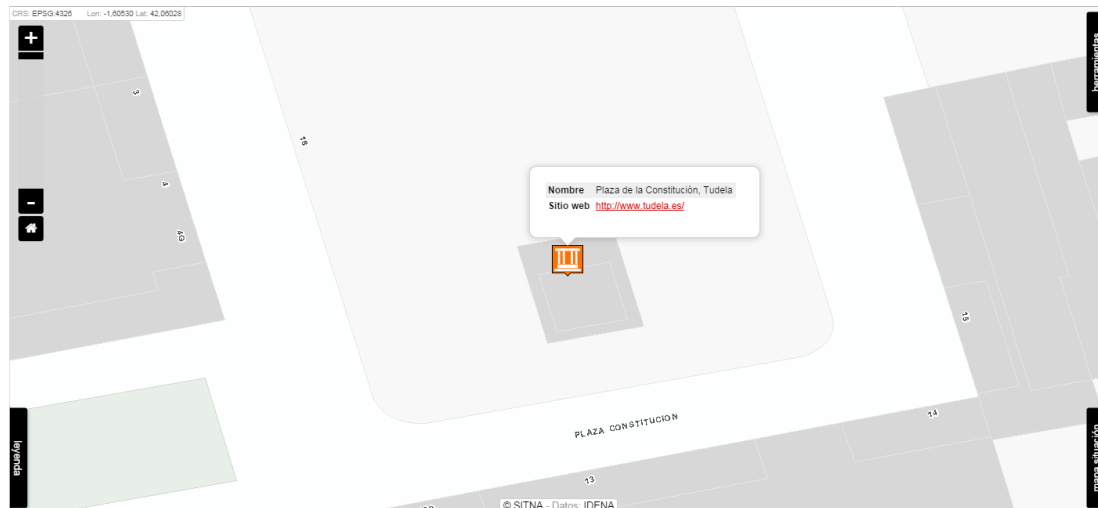


Figura 3: *Marcador con aspecto personalizado e información asociada.*

PERSONALIZACIÓN DEL VISOR

La configuración del mapa, su maquetación, su estilo y el conjunto de controles de usuario que se cargan pueden ser definidos mediante paquetes de maquetación, o *layouts*. Un *layout* es una carpeta localizada en una dirección web accesible al navegador del usuario final que contiene marcado HTML, código JavaScript, documentos CSS y documentos de configuración JSON que se inyectarán en la aplicación huésped (consumidora de la API) y que permiten personalizar el aspecto y comportamiento de los visores geográficos de manera sencilla, como se ve en el ejemplo de la figura 4.

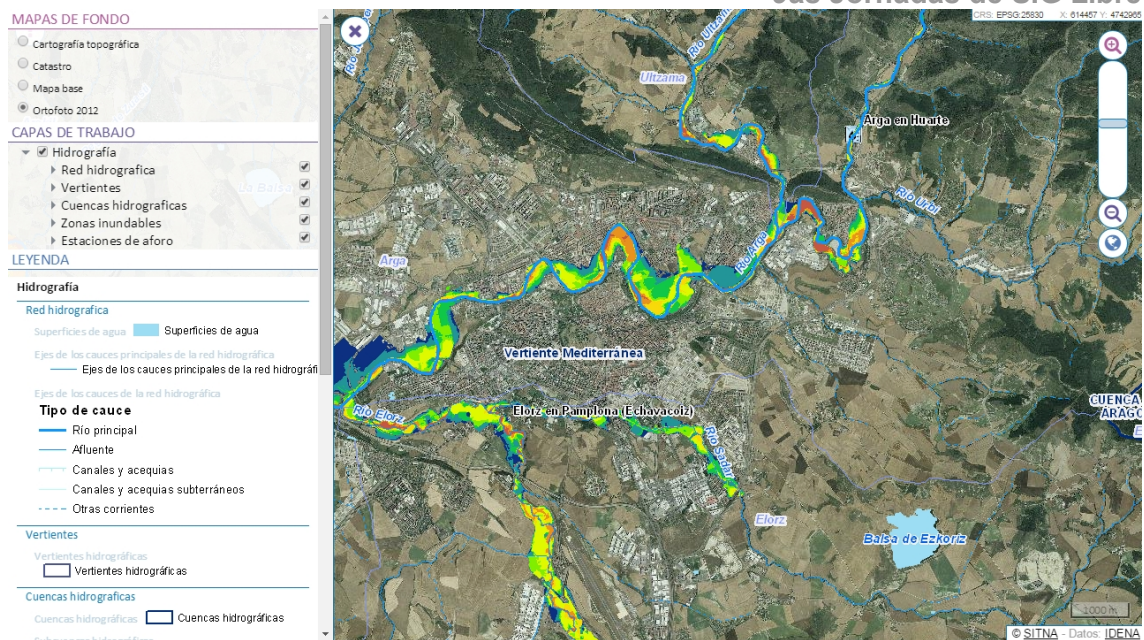


Figura 4: Visor con maquetaación personalizada.

El paquete de maquetaación por defecto tiene diseño adaptativo *mobile first*, es decir, pensado en primera instancia para pantallas de móviles y que adapta y enriquece la experiencia de usuario a medida que la pantalla gana tamaño y resolución.

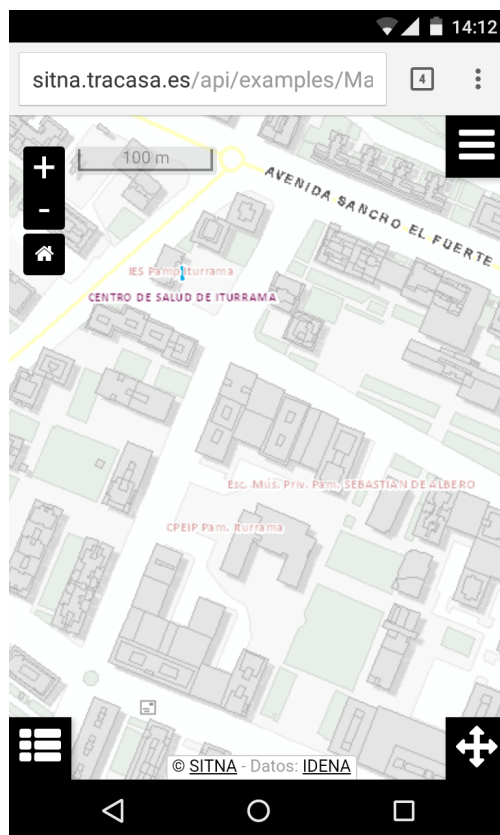


Figura 5: Aspecto de la maquetaación por defecto en un dispositivo móvil.

BIBLIOTECAS JAVASCRIPT

La API se apoya en varias bibliotecas de terceros para llevar a cabo sus funciones. No obstante, en aras de la facilidad de uso, el desarrollador no tiene que incluir ninguna referencia a esas bibliotecas en su aplicación web, tan solo tiene que incluir la referencia a la API, y es ésta la que se encarga de cargar los recursos según los va necesitando.

Para la navegación y el consumo y visualización de servicios OGC se utiliza OpenLayers; próximamente se podrá elegir por configuración entre la versión 2 y la nueva versión 3. Para asegurar la compatibilidad con dos versiones, que en realidad son dos APIs completamente distintas con jerarquías de objetos incompatibles, se utiliza un conjunto de clases envoltorio específico para cada versión, abstrayendo el resto del código de la API de la biblioteca geográfica utilizada. De esta manera cabría desarrollar un nuevo envoltorio para utilizar otra biblioteca como por ejemplo Leaflet en lugar de OpenLayers.

Dado que fue creada con la idea de cumplir unas necesidades muy concretas, la API SITNA ofrece solamente un subconjunto de las funcionalidades que ofrece OpenLayers, pero por otro lado tiene otras capacidades que exceden el ámbito de actuación de esta biblioteca geográfica, como es la mencionada maquetación personalizable o la creación de controles de usuario mediante plantilla.

EMBEBIDO DE MAPAS SIN DESARROLLO

Complementariamente a la API de desarrollo, se ha puesto a disposición del público una aplicación final de mapas [9] construida con ella y que acepta como parámetro la URL de un documento KML. La aplicación muestra los contenidos del KML en el mapa. El objetivo es que un diseñador web pueda incrustar en una página un visor de mapas que muestre la información que desee, simplemente enlazando a esta aplicación mediante un elemento HTML `IFRAME`, sin tener que realizar ningún tipo de desarrollo.

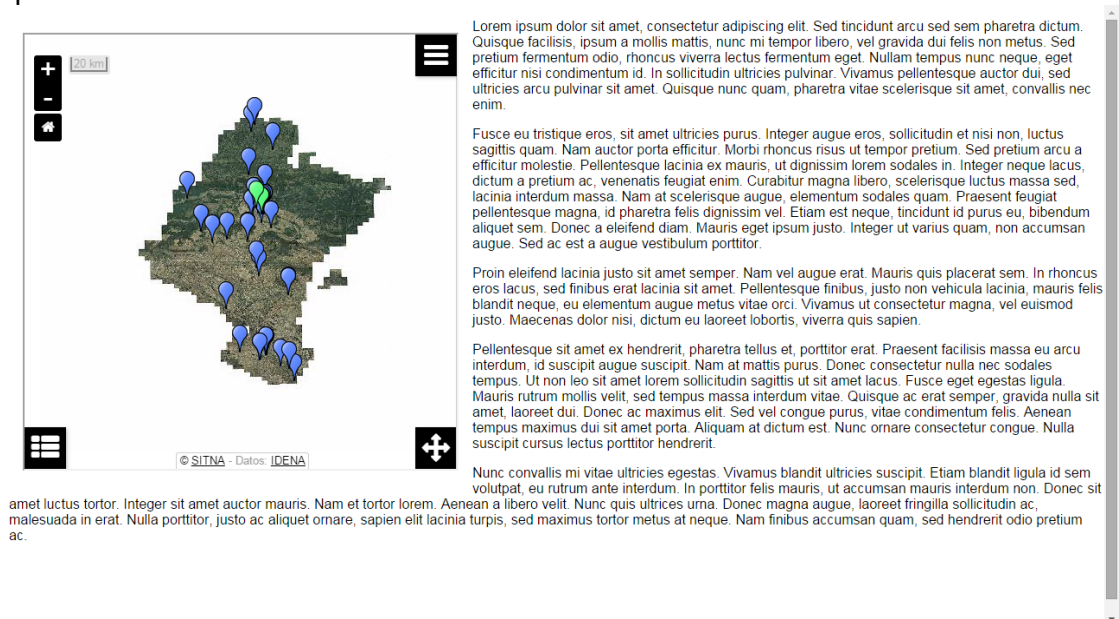


Figura 6: Mapa embebido en página HTML mediante elemento `IFRAME`.

USUARIOS POTENCIALES

Como se ha mencionado, la API SITNA se ha diseñado con los requisitos de desarrollo del Gobierno de Navarra. No obstante, es de acceso público, y cualquier tercero puede utilizarla para sus desarrollos. La licencia de uso es la BSD de 2 cláusulas [10].

CONCLUSIONES

El uso de la API SITNA está permitiendo a desarrolladores web sin experiencia GIS incrustar información con componente geográfico en sus aplicaciones HTML. Con ella ya se han realizado varios desarrollos con éxito. La figura 7 muestra una captura de pantalla de uno de esas aplicaciones.

Consulta Edificios

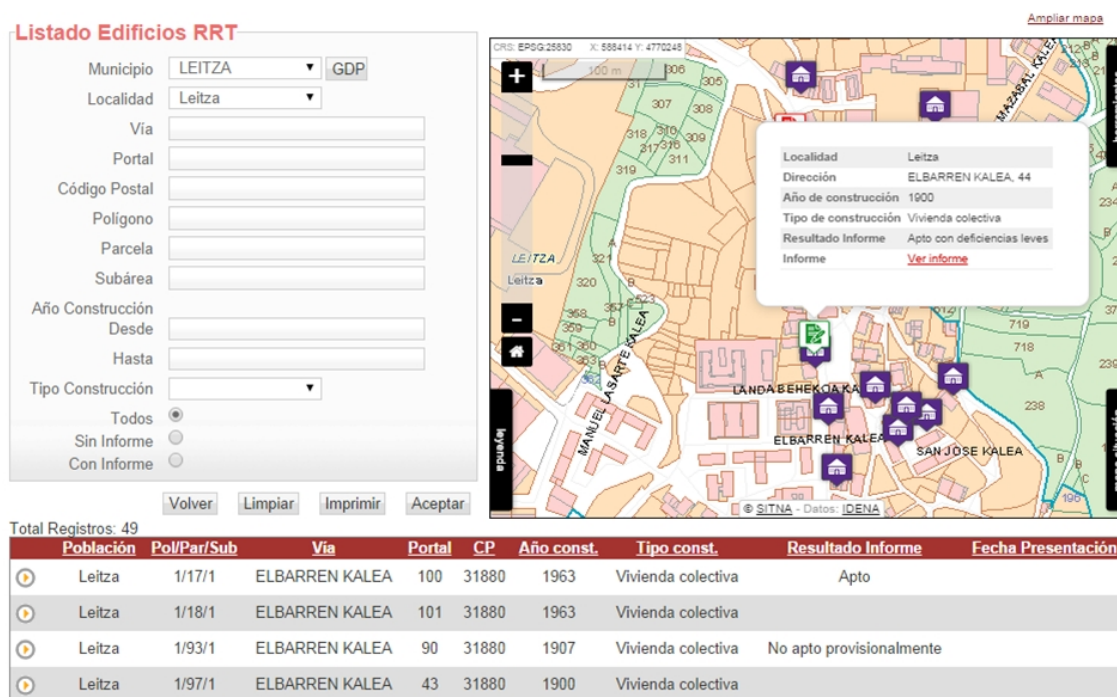


Figura 7: Aplicación con mapa incrustado basado en la API SITNA.

Incluir la API en los nuevos desarrollos está facilitando la reutilización de código y la minimización de errores de programación.

Finalmente, la compatibilidad de la API con OpenLayers 2 y 3 está permitiendo migrar sin esfuerzo la versión de aplicaciones existentes aunque estas hubieran sido desarrolladas antes de la publicación de la versión definitiva de OpenLayers 3.

AGRADECIMIENTOS

Han colaborado en la elaboración de este trabajo D. Juan Luis Cardoso Santos y D. Carlos Sabando Grasa, del área Geoweb del Departamento de Sistemas de Información Territorial de Tracasa.

REFERENCIAS

- ◆ [1] OGC: *Open Geospatial Consortium*, <http://www.opengeospatial.org/>
- ◆ [2] IDENA: *Infraestructura de Datos Espaciales de Navarra*, <http://idena.navarra.es/>
- ◆ [3] *OpenData - Portal de Gobierno Abierto de la Comunidad Foral de Navarra*, <http://www.gobiernoabierto.navarra.es/es/open-data>
- ◆ [4] Google - *API de Google Maps*, <https://developers.google.com/maps/>
- ◆ [5] *OpenLayers 3: A high-performance, feature-packed library for all your mapping needs*, <http://openlayers.org/>
- ◆ [6] *OpenLayers 2: Free maps for the web*, <http://openlayers.org/two/>
- ◆ [7] *Leaflet - a JavaScript library for mobile-friendly maps*, <http://leafletjs.com/>
- ◆ [8] *API SITNA*, <http://sitna.navarra.es/api/doc/>
- ◆ [9] *IFRAME de API SITNA*, <http://sitna.navarra.es/kml/>
- ◆ [10] *BSD 2-Clause License (FreeBSD/Simplified)* [https://tldrlegal.com/license/bsd-2-clause-license-\(freebsd\)](https://tldrlegal.com/license/bsd-2-clause-license-(freebsd))