

Consumo de APIs Java

Algunas aplicaciones como Google Maps o Facebook proporcionan su funcionalidad a través de una API JavaScript. Estas APIs ocultan todos los detalles de bajo nivel necesarios para interaccionar con las aplicaciones y facilitan su integración. A su vez, existen librerías que permiten trabajar con dichas APIs desde nuestras aplicaciones GWT usando código Java simplificando aún más la integración. En esta práctica, haremos uso de la API Java de GWT para Google Maps.

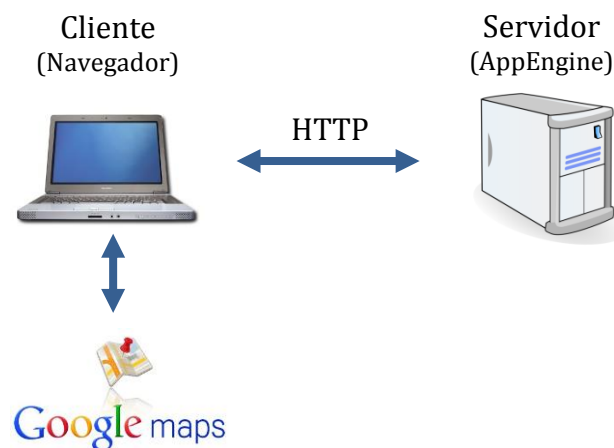


Figura 1. Esquema general de la práctica

Ejercicios

Ejercicio 1. Crear un mapa simple

Crear un mapa simple centrado en Sevilla (latitud = 37.35819, longitud: -5.98637). Use la clase "SimpleMap.java" para la implementación.

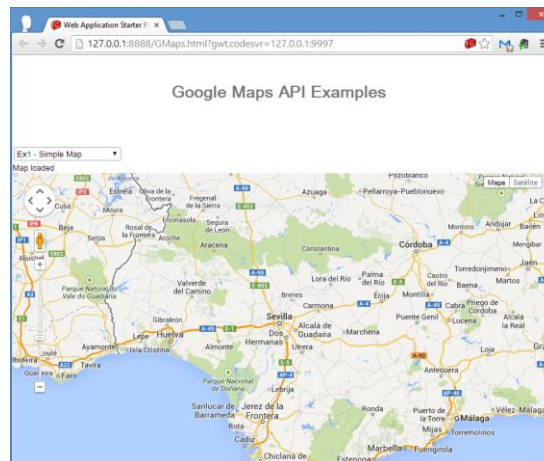


Figura 1. Mapa simple centrado en Sevilla

Ejercicio 2. Cargar mapa de forma asíncrona

Cree el mapa del ejercicio anterior pero esta vez cárguelo de forma asíncrona. Es decir, la aplicación no debería quedar bloqueada hasta mostrar el mapa. Use la clase “AsynchronousMap.java” para la implementación.

Ejercicio 3. Crear un mapa con controles personalizados

Crear un mapa centrado en Sevilla. El selector de tipo de mapa debe ser una lista desplegable. Además, el selector de zoom debe tener un tamaño pequeño y el mapa de mostrarse en vista satélite. Use la clase “CustomizedMap.java” para la implementación.

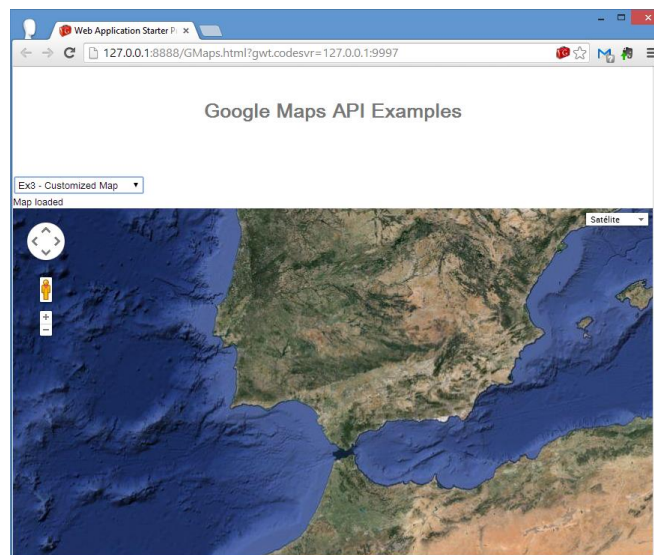


Figura 2. Mapa con controles personalizados

Ejercicio 4. Añadir un marcador al mapa.

Crear un mapa centrado en Sevilla como el descrito en los ejercicios anteriores. Incluir un marcador indicando la dirección de nuestra escuela (ETSII) y una ventana de información asociada que muestre una foto de la entrada principal. Use la clase “SchoolMap.java” para la implementación.

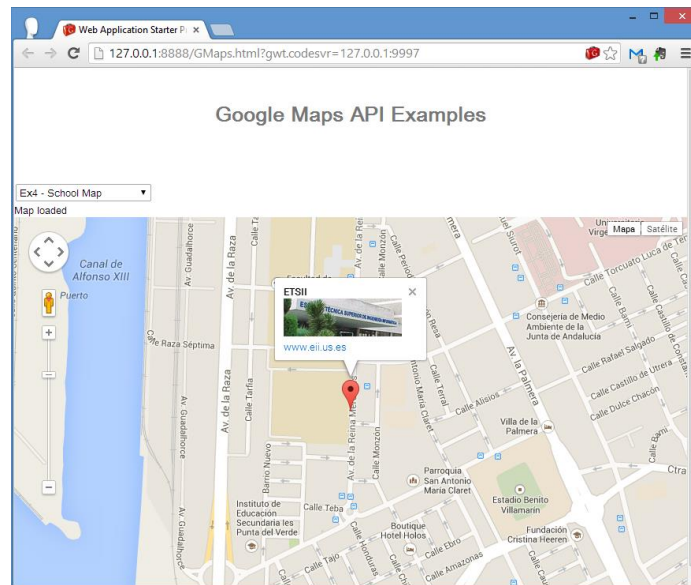


Figura 3. Mapa mostrando una marca y ventana de información asociada

Ejercicio 5. Geolocalización

Crear un mapa centrado en la posición actual del usuario. Use la clase “GeoMap.java” para la implementación.

Ejercicio 6. Búsqueda de direcciones

Crear un mapa centrado en Sevilla como el descrito en los ejercicios anteriores. Usar el objeto Geocoder de Google Maps para situar en el mapa la dirección indicada por el usuario. Use la clase “SearchMap.java” para la implementación

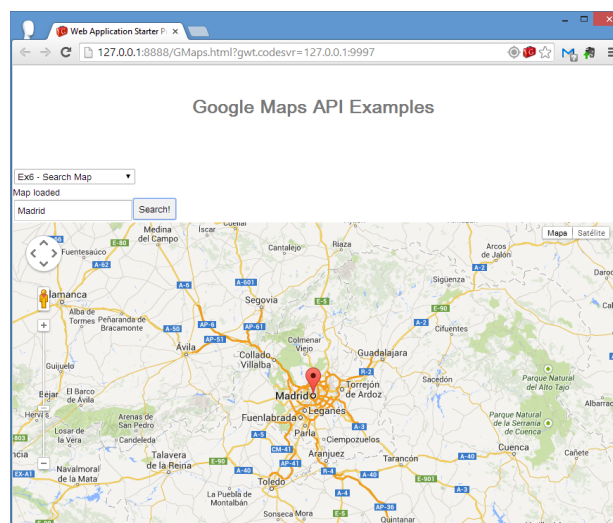


Figura 4. Mapa con búsqueda integrada

Ejercicios Extra

Ejercicio 7. Superponer datos geolocalizados

Crear un mapa centrado en Sevilla como el de los ejercicios anteriores. Una vez creado, añadir información sobre las arboledas singulares de la provincia. Dicha información puede localizarse en un fichero KML proporcionado por la consejería de medio ambiente cuya URI se proporciona. Use la clase “KmlMap.java” para la implementación.

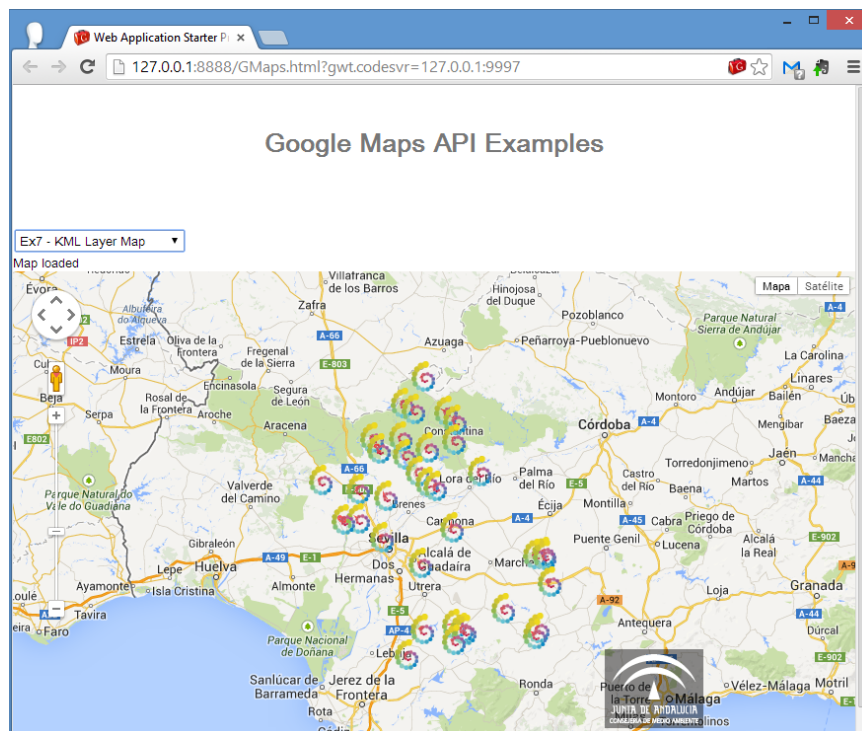


Figura 5. Superposición de información geográfica desde un fichero KML

Ejercicio 8. Desplegar la aplicación en AppEngine

Despliegue la aplicación en AppEngine y asegúrese de que todo funciona correctamente.

Ejercicio 9. Experimentar con los ejemplos de GMaps

Explorar los ejemplos proporcionados con la librería de GMaps para GWT y hacer nuevos tipos de mapas:

<http://code.google.com/p/gwt-google-apis/downloads/detail?name=gwt-maps-3.8.0-pre1.zip>

Ejercicio 10. Consumo de otras APIs Java para GWT

Existen otras APIs Java para GWT que permiten la integración con otras aplicaciones como Google+, Google Calendar o Google Latitude. El objetivo de este ejercicio es hacer uso de alguna de estas API y diseñar mashups que combinen su funcionalidad.

Listado de APIs para aplicaciones Google: <http://code.google.com/p/gwt-google-apis/wiki/Downloads>