



INSTITUTO POLITÉCNICO NACIONAL
ESCUELA SUPERIOR DE CÓMPUTO



Application Development For Mobile Devices

Semana 2

Emiliano González López

11/Junio/2020

Mapas

La API de Google Location Services, parte de servicios de Google Play, ofrece un marco más potente, de alto nivel que automatiza tareas como la elección del proveedor y la administración de energía. Los servicios de Localización también proporcionan nuevas características, como la detección de la actividad que no está disponible en la API. Los desarrolladores que utilizan la API, así como los que apenas están agregando la ubicación a las aplicaciones, deben considerar seriamente el uso de la API de servicios de localización.

La localización y las aplicaciones basadas en mapas ofrecen una experiencia atractiva en los dispositivos móviles. Se pueden construir estas capacidades en la aplicación utilizando las clases del paquete `android.location` y la API de Google Maps para Android.

Servicios de localización

Android le da a las aplicaciones el acceso a los servicios de localización, compatible con el dispositivo, a través de las clases del paquete `android.location`. El componente central de la ubicación es el servicio del sistema `LocationManager`, que proporciona APIs para determinar la ubicación y el manejo del dispositivo subyacente (si está disponible).

Con la API de Google Maps para Android, se pueden añadir mapas a la aplicación, que se basan en datos de mapas de Google. La API se encarga de automatizar el acceso a los servidores de Google Maps, la descarga de datos, visualización del mapa, y digitado de los gestos en el mapa. También se pueden utilizar llamadas a la API para agregar marcadores, polígonos y superposiciones, y para cambiar la vista del usuario de un mapa de una zona en particular.



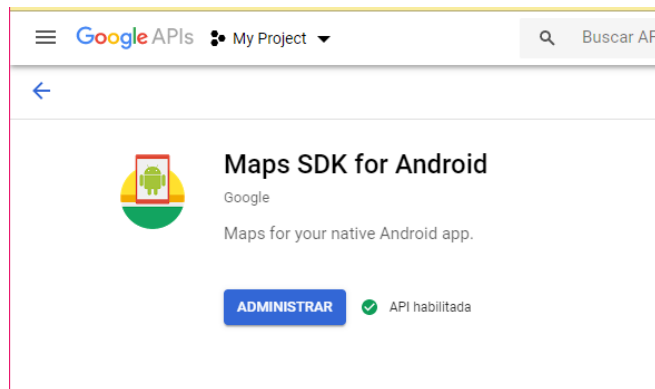
La clase de la clave de la API de Google Maps para Android es MapView. Un MapView muestra un mapa con los datos obtenidos del servicio de Google Maps. Cuando el MapView tiene el foco, se captarán las digitaciones y gestos táctiles para alejar o acercar el mapa de forma automática, incluyendo el manejo de las solicitudes de red para secciones de mapas adicionales. También proporciona todos los elementos necesarios para la interfaz de usuario para controlar el mapa. La aplicación también puede utilizar los métodos de clase MapView para controlar el mapa mediante programación y extraer una serie de superposiciones en la parte superior del mapa.

Desarrollo.

Ejercicio 1.

Crear una aplicación que provee soporte para ocupar la API de Google Maps en este caso tenemos que agregar el soporte para nuestra app, primeramente, iniciando sesión dentro del nuestro Android Studio, para después conectar la aplicación con un generador de Key, para poder ocupar esta API, ya que es un servicio de Google Cloud, y si en un futuro quisiéramos poner la app en la tienda de apps, se nos cobraría por el uso de la API.

La siguiente imagen muestra como el permiso de uso de Maps para mi aplicación está habilitado.



Código de implementación

```
LatLng geoPoint = new LatLng(540,1130);
Projection projection = mMap.getProjection();
projection.toScreenLocation(geoPoint);
Paint p = new Paint();
p.setColor(Color.BLUE);

btnSatelite.setOnClickListener((v) -> {
    if(mMap.getMapType() == GoogleMap.MAP_TYPE_SATELLITE)
        mMap.setMapType(GoogleMap.MAP_TYPE_NORMAL);
    else
        mMap.setMapType(GoogleMap.MAP_TYPE_SATELLITE);
});

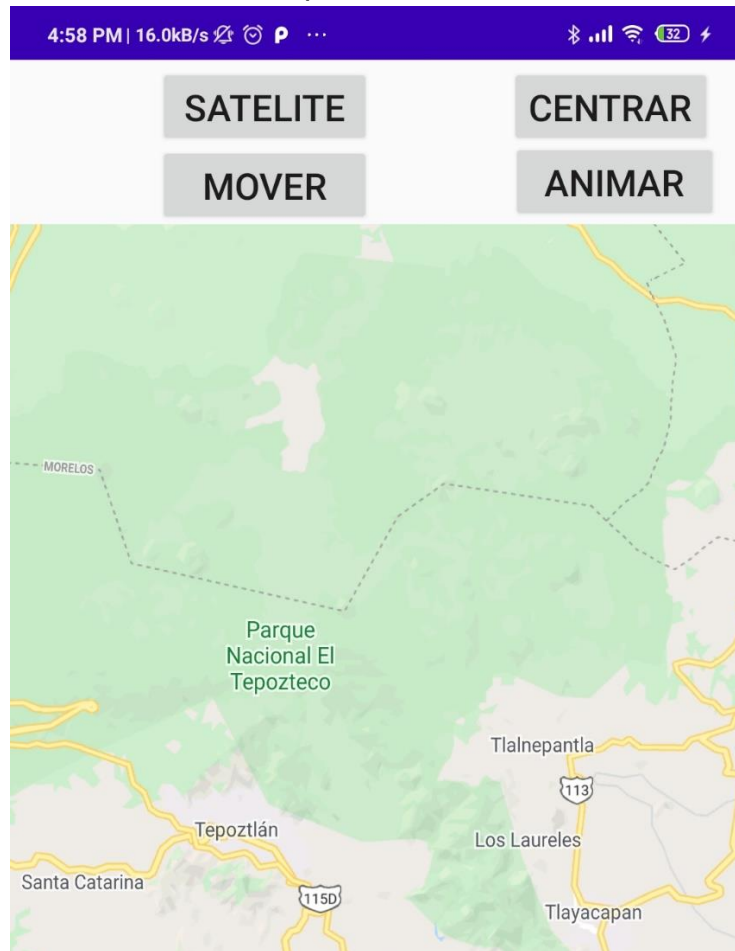
btnCentrar.setOnClickListener((v) -> {
    Double latitud = 37.40*1E6;
    Double longitud = -5.99*1E6;
    LatLng centerTo = new LatLng(latitud.intValue(), longitud.intValue());
    mMap.addMarker(new MarkerOptions().position(centerTo).title("Centrado"));
    mMap.moveCamera(CameraUpdateFactory.newLatLng(centerTo));
    mMap.moveCamera(CameraUpdateFactory.zoomTo(20));
});

btnAnimar.setOnClickListener((v) -> {
    Double latitud = 19.608415;
    Double longitud = -99.2502285;
    LatLng location = new LatLng(latitud, longitud);
    mMap.moveCamera(CameraUpdateFactory.newLatLngZoom(location, 20));
    String msg = "Lat: " + latitud + "-" + "Lon: " + longitud;
    Toast toast = Toast.makeText(MapsActivity.this, msg, Toast.LENGTH_SHORT);
    toast.show();
});
```

La API de Google ha cambiado con el tiempo por lo que hay algunas cosas que son diferentes como la creación de un punto geográfico, ahora se declara un objeto `LatLng`, o las animaciones, en general toda la API ha sido cambiada ya que las funciones ahora se llaman diferente o se encuentran ubicadas en diferentes clases.

En cualquier caso, la aplicación queda de la siguiente manera.

Foto de la aplicación funcionando.



En cuanto a lo que se refiere del overlay me fue imposible hacer que funcionara.

Conclusiones

El desarrollo de esta aplicación fue relativamente sencillo en la primer parte, pero la segunda de los overlays fue complicada ya que no logre hacer que funcionara, en algunas aplicaciones pasadas ya había hecho algo parecido pero nunca con el método de usar el canvas sobre él, por lo que fue bastante difícil ya que no entendí bien como es que funcionaba ese método.

Referencias

<https://developers.google.com/maps/documentation/android-sdk/intro>