



INSTITUTO POLITÉCNICO NACIONAL
ESCUELA SUPERIOR DE CÓMPUTO



Application Development For Mobile Devices

Semana 2

Emiliano González López

11/Junio/2020

Geolocalización

En un móvil, la localización usualmente es con el GPS y con la red telefónica. Con el método `getAllProviders()` de la clase `LocationManager` se realiza la localización. Con una instancia del administrador de localización `getSystemService(LOCATION_SERVICE)` se obtiene la lista de proveedores del servicio

La clase `LocationManager` posee el método `getLastKnownLocation(String provider)`, que devuelve la última posición que se obtuvo a través del proveedor desde su uso.

Se debe suscribir al evento que se lanza cuando un proveedor recibe nuevos datos sobre la localización actual. Se le deben pasar cuatro parámetros al método `requestLocationUpdates()`:

- Nombre del proveedor de localización al que se desea suscribir.
- Tiempo mínimo, en milisegundos, entre actualizaciones.
- Distancia mínima, en metros, entre actualizaciones.
- Una instancia del objeto `LocationListener`, para definir las acciones de actualización de la posición.

Desarrollo.

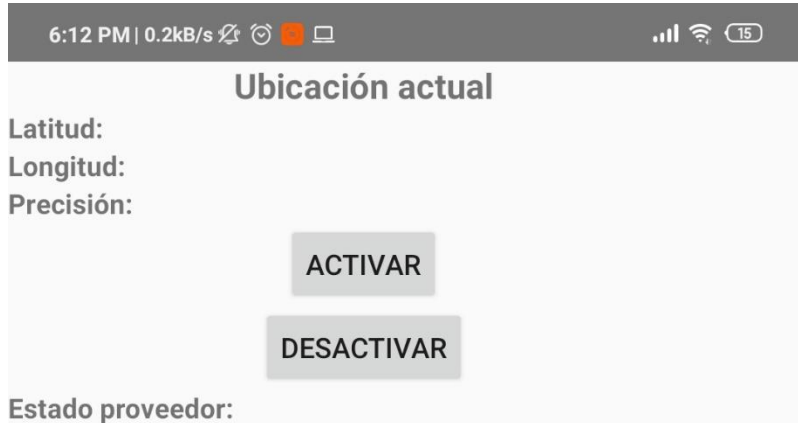
Ejercicio 1

El primer ejercicio, nos pide obtener los proveedores de localización para así obtener la latitud y la longitud y desplegarlo en pantalla.

En esta practica tuvo varios problemas ya que la forma de hacer las cosas ha cambiado un poco, pero, después de investigar un poco llegue a como poder resolver los problemas, estoy fueron agregando alguna líneas que verificaran el estado de los permisos y nuestra versión de Android.

```
try {
    Criteria criteria = new Criteria();
    criteria.setAccuracy(1);
    criteria.setAltitudeRequired(true);
    this.provider1 = this.locationManager.getBestProvider(criteria, enabledOnly: false);
    if (this.locationManager.isProviderEnabled(this.provider1)) {
        this.provider = this.locationManager.getProvider(this.provider1);
        this.precision = this.provider.getAccuracy();
        this.altitud = this.provider.supportsAltitude();
        this.consumoRecursos = this.provider.getPowerRequirement();
        this.status = 1;
        this.location = this.locationManager.getLastKnownLocation(this.provider1);
        if (this.location != null) {
            Log.d( tag: "iniciarUbicacion:", msg: "Location NO es nulo");
            this.currentLocation = new Latlng(this.location.getLatitude(), this.location.getLongitude())
            return;
        }
        Log.d( tag: "iniciarUbicacion:", msg: "Location es nulo");
        return;
    }
    Log.d( tag: "iniciarUbicacion", msg: "No encuentre el mejor proveedor");
} catch (Exception e) {
    Log.d( tag: "iniciarUbicacion:", msg: "Error en el try");
    e.printStackTrace();
}
} else {
    Log.d( tag: "initLocationService", msg: "No hay permiso de algo");
    requestPermissions(new String[]{"android.permission.ACCESS_FINE_LOCATION", "android.permission.ACCESS_COARSE_LOCATION"}, 1);
}
```

Las siguientes dos imágenes muestran como se ejecuta la aplicación al abrirá y al activar la geolocalización



Conclusiones

Esta práctica fue compleja relativamente, ya que nunca había trabajado con esta clase, ni hecho algún ejercicio con la geolocalización desde el celular, únicamente recuperando los datos desde un sensor y base de datos Firebase, una vez más estuvo presente el problema de compatibilidad pero no fue realmente difícil de solucionar ya que encontré una marea sencilla de solucionar con el solucionador del IDE, siendo esta agregar una línea que verifique que nuestra versión de Android sea superior a cierta versión en este caso la M.

Referencias

<https://developer.android.com/reference/android/location/LocationManager>

<https://developer.android.com/training/permissions/requesting>

<https://www.geeksforgeeks.org/android-how-to-request-permissions-in-android-application/>