

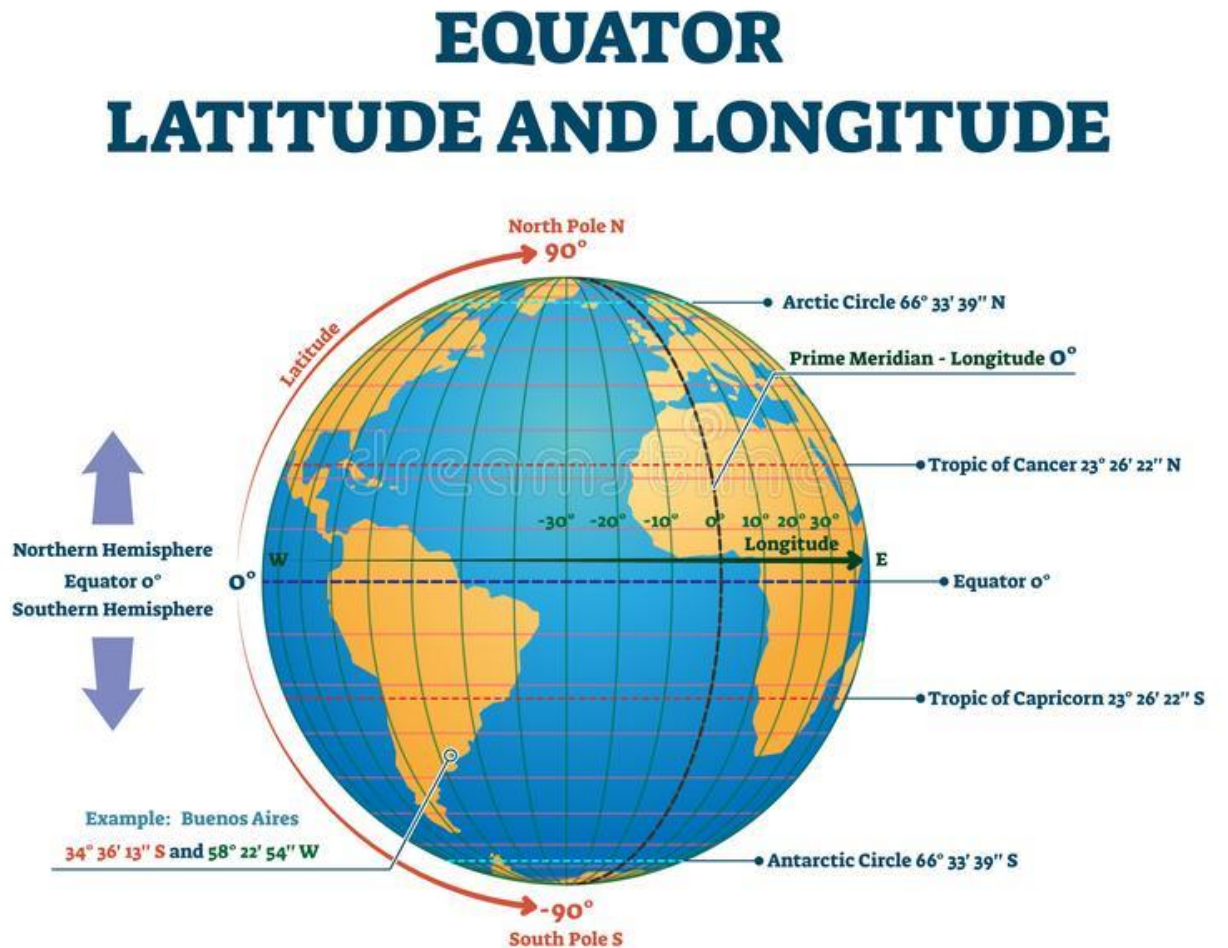
Pert 13: GPS

Introduction

- Getting user's current location is the most important task in apps those provides services like food ordering, transportation, health tracking, social networking or lot more.
- This location information mainly based on getting user's latitude & longitude. Using latitude & longitude values, you can easily determine the exact location of a user.
- It may seem a lot of programming efforts on doing this, but thankfully the android framework itself provides a location API called Fused Location Provider to shorten our job.

Lat Long

- Latitude and longitude are angles that uniquely define points on a sphere.
- Together, the angles comprise a coordinate scheme that can locate or identify geographic positions on the surfaces of planets such as the earth.



The fused location provider

- The fused location provider retrieves the device's last known location. The fused location provider is one of the location APIs in Google Play services.
- It manages the underlying location technology and provides a simple API so that you can specify requirements at a high level, like high accuracy or low power. Some advantages of using this API are:
 - It provides simple and easy to use APIs.
 - Provides high accuracy over other options.
 - Utilizes low power by choosing the most efficient way to access the location.

Lab 1: PosisiSaatIni

- Buatlah project : PosisiSaatIni
- Fungsi Project: menampilkan data lokasi saat ini
- Konfigurasi Gradle

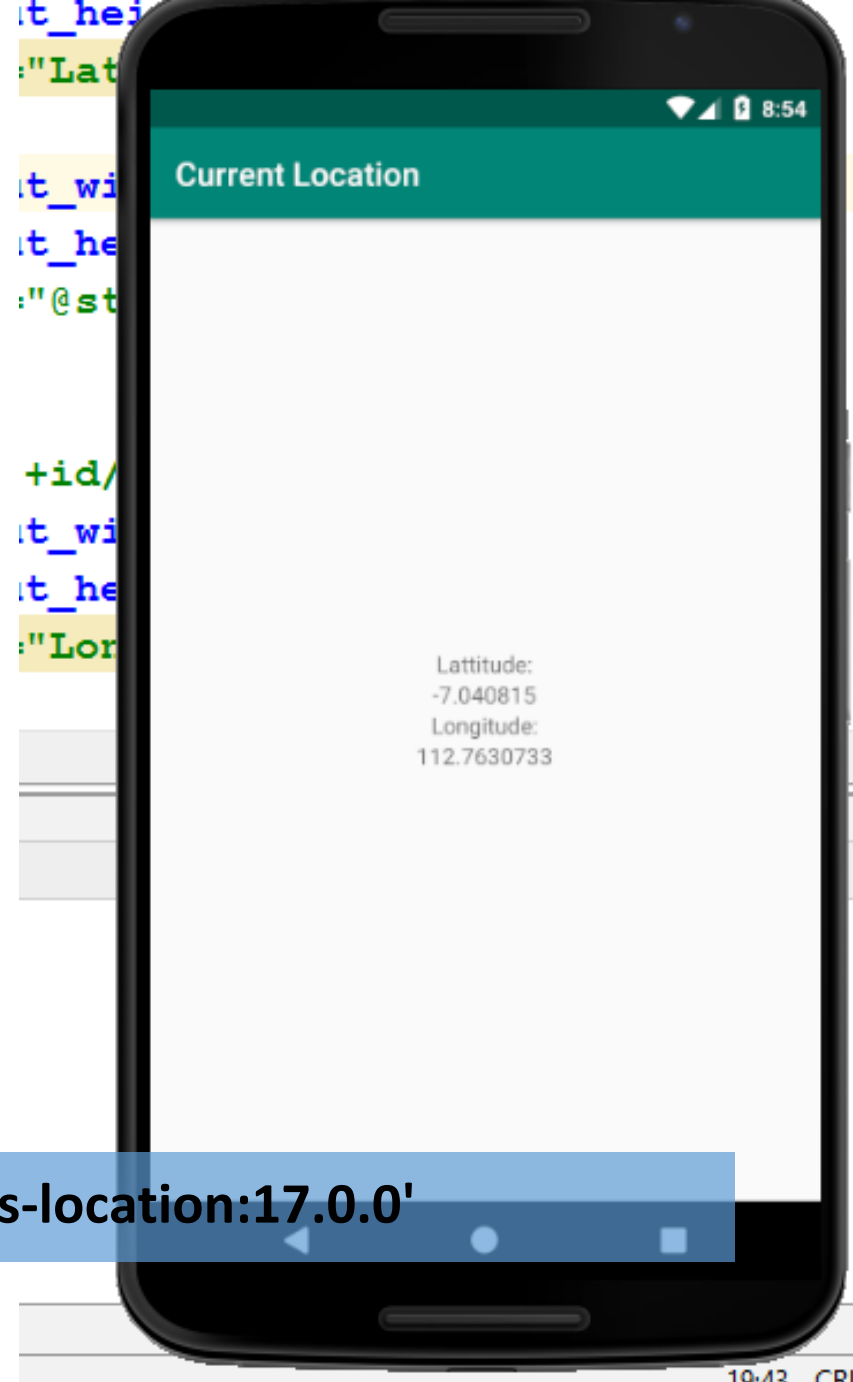
- Project

```
repositories {  
    google()  
    jcenter()  
}
```

- App

```
//.....  
//.....
```

```
implementation 'com.google.android.gms:play-services-location:17.0.0'
```



Manifest

```
<uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION" />  
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
```

- Tambahkan konfigurasi permission

Method untuk check permission

```
private boolean checkPermissions(){  
    if (ActivityCompat.checkSelfPermission(this,  
        Manifest.permission.ACCESS_COARSE_LOCATION) ==  
        PackageManager.PERMISSION_GRANTED &&  
        ActivityCompat.checkSelfPermission(this,  
            Manifest.permission.ACCESS_FINE_LOCATION) ==  
            PackageManager.PERMISSION_GRANTED){  
        return true;  
    }  
    return false;  
}
```

Method untuk meminta permission

```
private void requestPermissions(){  
    ActivityCompat.requestPermissions(  
        this,  
        new String[]{Manifest.permission.ACCESS_COARSE_LOCATION,  
Manifest.permission.ACCESS_FINE_LOCATION},  
        PERMISSION_ID  
    );  
}
```


Method untuk menangani hasil permintaan permission

@Override

```
public void onRequestPermissionsResult(int requestCode, String[] permissions, int[] grantResults) {  
    super.onRequestPermissionsResult(requestCode, permissions, grantResults);  
    if (requestCode == PERMISSION_ID) {  
        if (grantResults.length > 0 && grantResults[0] ==  
PackageManager.PERMISSION_GRANTED){  
            // Granted. Start getting the location information  
        }  
    }  
}
```

Check apakah setting Location enable

```
private boolean isLocationEnabled(){  
    LocationManager locationManager = (LocationManager)  
    getSystemService(Context.LOCATION_SERVICE);  
    return  
    locationManager.isProviderEnabled(LocationManager.GPS_PROVIDER)  
    || locationManager.isProviderEnabled(  
        LocationManager.NETWORK_PROVIDER  
    );  
}
```