In this chapter we are going to learn about location based applications and Google maps integration.
Configure google services

Device

Google Play services library

Your app

Google API Client

Google Play services

Games service

Drive service

Other services
Configure google services– step1

- In the SDK manager, add these extras:
Configure google services – step 2

- Update the dependencies and press ‘sync’
Configure google services – step 3

- Add the `<meta-data>` to the manifest

```
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
package="sheep.com.AnimalFestival" >

<application>
<allowBackup="true"
android:icon="@drawable/ic_launcher"
android:label="AnimalFestival"
android:theme="@style/AppTheme">

<meta-data android:name="com.google.android.gms.version"
android:value="@integer/google_play_services_version" />

<activity>

android:name=".MainActivity"
android:label="AnimalFestival">
<intent-filter>
<action android:name="android.intent.action.MAIN" />
</intent-filter>
</activity>

</application>

</manifest>
```
Adding permissions

- City block accuracy

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

- GPS accuracy

<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION"/>
Location sources

- GPS
- Cellular Network
- Wi-Fi
GPS

- Accurate up to few meters
- High battery consumption
- Outdoor
- Slow Response

Tour – Google Maps
Cellular Network

- from 20m to 2000m  Accuracy
- Low battery consumption
- Almost always available
- Indoor
- Fast response
Wi-Fi

- from 20m to 80m Accuracy
- Low battery consumption
- Indoor
- Fast response
Fused Location

- App
  - Other Applications
  - GPS
  - Network
  - Sensors
  - Wi-Fi
In most cases you can get the location from the device in 0 time (*)

```java
mCurrentLocation = LocationServices.FusedLocationApi.getLastLocation(mGoogleApiClient);
```
Importing location project – Step 1

- Get the projects code from google git:

  https://github.com/googlesamples/android-play-location/tree/master/LocationUpdates
Importing location project—Step 2

- Extract the zip file and import the project from Android Studio:
Checking Distance between locations

Location location1 = ...;
Location location2 = ...;

float distanceInMeters =
  location1.distanceTo(location2);
Ex15 - Geo-Fencing

- Get indication when we get into a certain area
- Advanced: Show how close we are
- Very Advanced: The user can enter the address instead of lat/lon
Ex15 - Geo-Fencing

Please enter lat
Please enter lon
Please enter accuracy

START  STOP
How to incorporate Google Maps into your application?

https://developers.google.com/maps/documentation/android/start