

## Demonstration Guide:

Before starting, create a new blank project in AndroidStudio, or follow the explanation with the available code at (some github link).

### 1. Dependencies:

#### a. Gradle:

- i. On the build.gradle of our project, we need to add the following dependencies:
  1. `implementation("com.gorisse.thomas.sceneform:sceneform:1.20.1")` - this dependence fetches the Sceneform Maintained resources.

#### b. AndroidManifest:

- i. On the AndroidManifest of our project, add the following lines:
  1. Before the `<application>` header:
    - a. `<uses-permission android:name="android.permission.CAMERA" />`
    - b. `<uses-permission android:name="android.permission.INTERNET" />`
    - c. The first will add camera access to the necessary permissions of our app and the second allows internet communications.
  2. After the `<application>` header:
    - a. `<meta-data android:name="com.google.ar.core" android:value="optional" />`
    - b. If true it would require that "Google Play Services for AR" are installed, since the app doesn't contain any features that aren't related to AR.

### 2. Layouts

#### a. Activity\_main

- i. Delete the default TextView and define a FragmentContainer View. Give it the id `ar_fragment`:
  1. `android:id="@+id/ar_fragment"`
- ii. Layout width and height should be `"match_parent"`
- b. `View_model_text` - this layout shows a title above the model.
  - i. Create a new layout, delete the default layout and define a TextView with the text `"Tiger"`

### 3. MainActivity

- a. Class has to implement  
FragmentOnAttachListener, BaseArFragment.OnTapArPlaneListener  
BaseArFragment.OnSessionConfigurationListener,  
ArFragment.OnViewCreatedListener
  - b. Define:
    - i. `private ArFragment arFragment;`
    - ii. `private Renderable model;`
    - iii. `private ViewRenderable viewRenderable;`
  - c. Define onAttachFragment, which sets the  
SessionConfigurationListener and the OnTapArPlaneListener  
on the arFragment.
  - d. Define onSessionConfiguration to allow depth.
  - e. Define onViewCreated, setFrameRateFactor to limit frame  
rate.
  - f. Define onTapPlane, which loads the model, sets the anchor  
point, defines the model and links it to the view.
  - g. Define loadModels, which sets the model on the  
MainActivity and renders the view with the model.
4. Demonstration
- a. Skip this step if the requirements are already met.
    - i. Install a Pixel 4 running API 30 of Android or
    - ii. Use a Android device with Android 11 or higher
  - b. Update Google Play Services and download Google AR Core  
from the Play Store if necessary.
  - c. Install the application in the device and run.