

# AR LAB GUIDE

- **Creating our first AR app.**

- Open a terminal and navigate to the folder where you want to create your app (in my case, *CordovaProjects*):

```
$ cd ~/CordovaProjects/
```

- Create a project running the following command:

```
$ cordova create ARFirstApp com.yourid.arsample ARFApp
```

- Add the platforms that you want to create the app for:

```
$ cordova platforms add android@6.2.3 (depends on version)
```

```
$ cordova platforms add ios
```

- Add Cordova plugin (it takes a while):

```
$ cordova plugin add https://github.com/Wikitude/wikitude-cordova-plugin.git
```

```
cordova plugin add cordova-plugin-compat
```

```
cordova plugin add cordova-plugin-file
```

- You can check if everything has been added by typing:

```
$ cordova plugin ls
```

```
$ cordova platforms ls
```

Here you can see the code that (more or less) we are going to be working with. It's here in case you need to check, but it should be enough with following the explanation.

```
var worlds = [  
  {  
    "path": "www/world/1st/index.html",  
    "requiredFeatures": [  
      "geo" // We could have image_tracking too  
    ],  
    "startupConfiguration": {  
      "camera_position": "back", // back camera  
      "camera_resolution": "auto" // automatic will do  
    }  
  }  
]
```

```

    }
]
function getWorldPath(index) {
    return worlds[index];
}

```

---

```

var world = {
    loaded: false, // Checks if world has been already loaded
    loadPois: loadPoisFn(poiData) {
        var image = new AR.ImageResource('img/marker.png');
        var imageDrawable = new AR.ImageDrawable(image, 2.5, {
            opacity: 0.9 // Doesn't matter actually
        });
        var location = new AR.GeoLocation(poiData.lat,
            poiData.lon, poiData.alt);
        var geoObject = new AR.GeoObject(location, {
            cam: [imageDrawable]
        });
    },
    locationChanged: function onLocationChanged(lat, lon, alt,
        acc) {
        if (!world.loaded) {
            poiData = {
                "lat": lat + (Math.random()/5 - 0.1), // randomness
                "lon": lon + (Math.random()/5 - 0.1),
                "alt": alt
            };
            world.loadPois(poiData);
            world.loaded = true;
        }
    }
}

AR.context.onLocationChanged = world.locationChanged;

```

---

```
console.log('alt: ' + poiData.alt + ' lon: ' + poiData.lon);
```

---

```
$ cordova run android
```

---

## ERRORS:

\* What went wrong:

A problem occurred configuring root project 'android'.

> You have not accepted the license agreements of the following SDK components:

[Android SDK Platform (number)].

Before building your project, you need to accept the license agreements and complete the installation of the missing components using the Android Studio SDK Manager. Alternatively, to learn how to transfer the license agreements from one workstation to another, go to <http://d.android.com/r/studio-ui/export-licenses.html>

---

Manifest merger failed:

uses-sdk: minSdkVersion 16 cannot be smaller than version 19 declared in library

---

```
var world = {
    imageDrawable: null,
    distance: null,
    ...
    loadPois: loadPoisFn(poiData) {
        world.imageDrawable = new AR...
        world.distance = new AR...
        ...
    changeSizeInverse: function changeSizeInv() {
        var dist = world.location.distanceToUser();
        var scale = 10/(dist + 1); // 9 meters = scale 1
        world.imageDrawable.scale = scale;
    },
    locationChanged: function onLocationChanged(...) {
        if (!world.loaded) {
            ...
        }
    }
}
```

```
    }  
    world.changeSizeInverse();  
}  
}
```

---

```
changeSizeLinear: function changeSizeLin() {  
    var dist = world.location.distanceToUser();  
    var scale = -0.1*dist + 10; // parameters depend on aug  
    world.imageDrawable.scale = scale;  
}
```