



UNIVERSITAT POLITÈCNICA DE CATALUNYA  
BARCELONATECH

---

Escola Superior d'Enginyeries Industrial,  
Aeroespacial i Audiovisual de Terrassa

---

Projecte de desenvolupament d'una  
aplicació per a l'aprenentatge del  
funcionament d'una màquina de taller  
amb ulleres de realitat augmentada

---

**Autor TFG:** David Alfaro Pérez

**Director TFG:** Xavier Salueña Berna

**Titulació:** Grau en Enginyeria Electrónica Industrial i Automàtica

**Data de lliurament:** 22/06/2021

# Annexes

## Índex

1. Codi de l'aplicació	2
1.1 build.gradle	2
1.2 AndroidManifest	3
1.3 MainActivity	4
1.4 SegonActivity	8
1.5 TercerActivity	11
1.6 QuartActivity	15

# 1. Codi de l'aplicació

## 1.1 build.gradle

```
apply plugin: 'com.android.application'
```

```
android {  
    compileSdkVersion 30  
    buildToolsVersion "30.0.2"  
  
    defaultConfig {  
        applicationId "com.example.tfgapp1"  
        minSdkVersion 22  
        targetSdkVersion 30  
        versionCode 1  
        versionName "1.0"  
  
        testInstrumentationRunner "androidx.test.runner.AndroidJUnitRunner"  
    }  
  
    buildTypes {  
        release {  
            minifyEnabled false  
            proguardFiles getDefaultProguardFile('proguard-android-optimize.txt'), 'proguard-rules.pro'  
        }  
    }  
}
```

```
dependencies {  
    implementation fileTree(dir: "libs", include: ["*.jar"])  
    implementation (name: 'wikitude-native-sdk', ext:'aar')  
    implementation "com.google.ar:core:1.1.0"  
    implementation 'androidx.appcompat:appcompat:1.3.0'  
    testImplementation 'junit:junit:4.12'  
    androidTestImplementation 'androidx.test.ext:junit:1.1.2'  
    androidTestImplementation 'androidx.test.espresso:espresso-core:3.3.0'  
    implementation 'com.google.android.material:material:1.0.0'  
    implementation 'com.google.android:flexbox:1.1.0'  
    implementation 'androidx.legacy:legacy-support-v4:1.0.0'  
    implementation 'androidx.legacy:legacy-support-v13:1.0.0'  
    implementation 'androidx.coordinatorlayout:coordinatorlayout:1.0.0'  
    implementation 'androidx.constraintlayout:constraintlayout:1.1.3'  
    implementation 'com.android.support.constraint:constraint-layout:1.1.3'
```

```
implementation 'com.google.android:flexbox:1.1.0'  
implementation "androidx.appcompat:appcompat:1.0.0"  
  
}  
repositories {  
    flatDir{  
        dirs 'libs'  
    }  
}
```

## 1.2 AndroidManifest

```
<manifest xmlns:android="http://schemas.android.com/apk/res/android"  
    package="com.example.tfgapp1">  
    <uses-permission android:name="android.permission.CAMERA" />  
    <uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />  
    <uses-permission android:name="android.permission.INTERNET" />  
  
    <uses-feature android:glEsVersion="0x00020000" android:required="true" />  
    <uses-feature android:name="android.hardware.camera" android:required="true" />  
  
    <application  
        android:allowBackup="true"  
        android:icon="@mipmap/ic_launcher"  
        android:label="@string/app_name"  
        android:roundIcon="@mipmap/ic_launcher_round"  
        android:supportsRtl="true"  
        android:theme="@style/AppTheme" >  
        <activity  
            android:name="com.example.tfgapp1.external.MainActivity"  
            android:configChanges="orientation|keyboardHidden|screenSize" >  
            <intent-filter>  
                <action android:name="android.intent.action.MAIN" />  
  
                <category android:name="android.intent.category.LAUNCHER" />  
            </intent-filter>  
        </activity>  
        <activity  
            android:name="com.example.tfgapp1.external.SegundoActivity"  
            android:configChanges="orientation|keyboardHidden|screenSize" >  
            <intent-filter>  
                <action android:name="android.intent.action.SEGUNDO" />  
  
                <category android:name="android.intent.category.LAUNCHER" />  
            </intent-filter>  
        </activity>  
    </application>  
</manifest>
```

```
</activity>
<activity
  android:name="com.example.tfgapp1.external.TercerActivity"
  android:configChanges="orientation|keyboardHidden|screenSize" >
  <intent-filter>
    <action android:name="android.intent.action.TERCERO" />

    <category android:name="android.intent.category.LAUNCHER" />
  </intent-filter>
</activity>
<activity
  android:name="com.example.tfgapp1.external.CuartoActivity"
  android:configChanges="orientation|keyboardHidden|screenSize" >
  <intent-filter>
    <action android:name="android.intent.action.CUARTO" />

    <category android:name="android.intent.category.LAUNCHER" />
  </intent-filter>
</activity>
</application>

</manifest>
```

## 1.3 MainActivity

```
package com.example.tfgapp1.external;

import com.wititude.NativeStartupConfiguration;
import com.wititude.WititudeSDK;
import com.wititude.common.WititudeError;
import com.wititude.common.camera.CameraSettings;
import com.wititude.common.rendering.RenderExtension;
import com.wititude.rendering.ExternalRendering;
import com.example.tfgapp1.external.render.CustomSurfaceView;
import com.example.tfgapp1.external.render.Driver;
import com.example.tfgapp1.external.render.GLRenderer;
import com.example.tfgapp1.external.render.StrokedRectangle;
import com.example.tfgapp1.external.render.DropDownAlert;
import com.wititude.tracker.ImageTarget;
import com.wititude.tracker.ImageTracker;
import com.wititude.tracker.ImageTrackerListener;
```

```
import com.wititude.tracker.TargetCollectionResource;
import android.app.Activity;
import android.content.Intent;
import android.os.Bundle;
import android.util.Log;
import android.view.WindowManager;
import android.view.View;

public class MainActivity extends Activity implements ImageTrackerListener,
ExternalRendering {

    private static final String TAG = "SimpleImageTracking";
    private WikitudeSDK wikitudeSDK;
    private CustomSurfaceView customSurfaceView;
    private Driver driver;
    private GLRenderer glRenderer;
    private DropDownAlert dropDownAlert;
    private DropDownAlert dropDownAlert2;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);

        wikitudeSDK = new WikitudeSDK(this);
        NativeStartupConfiguration startupConfiguration = new NativeStartupConfiguration();
        startupConfiguration.setLicenseKey(WaitedSDKSConstants.WAITED_SDKS_KEY);
        startupConfiguration.setCameraPosition(CameraSettings.CameraPosition.BACK);
        startupConfiguration.setCameraResolution(CameraSettings.CameraResolution.AUTO);

        wikitudeSDK.onCreate(getApplicationContext(), this, startupConfiguration);
        final TargetCollectionResource targetCollectionResource =
wikitudeSDK.getTrackerManager().createTargetCollectionResource("file:///android_asset/tracker_1.wtc");
        wikitudeSDK.getTrackerManager().createImageTracker(targetCollectionResource, this,
null);

        dropDownAlert2 = new DropDownAlert(this);
        dropDownAlert2.setText("Pas 1: Prem el botó blanc per iniciar la màquina CNC, si es
poden obrir les portes fés click en següent");
        dropDownAlert2.setTextWeight(0.5f);
        dropDownAlert2.show();
        getWindow().addFlags(WindowManager.LayoutParams.FLAG_KEEP_SCREEN_ON);
    }

    public void Instru (View view)
    {
        Intent Instru2 = new Intent( this, SegundoActivity.class);
        startActivity(Instru2);
    }
}
```

```
}

@Override
protected void onResume() {
    super.onResume();
    wiktitudeSDK.onResume();
    customSurfaceView.onResume();
    driver.start();
}

@Override
protected void onPause() {
    super.onPause();
    customSurfaceView.onPause();
    driver.stop();
    wiktitudeSDK.onPause();
}

@Override
protected void onDestroy() {
    super.onDestroy();
    wiktitudeSDK.onDestroy();
}

@Override
public void onRenderExtensionCreated(final RenderExtension renderExtension) {
    glRenderer = new GLRenderer(renderExtension);

wiktitudeSDK.getCameraManager().setRenderingCorrectedFovChangedListener(glRenderer)
;
    customSurfaceView = new CustomSurfaceView(getApplicationContext(), glRenderer);
    driver = new Driver(customSurfaceView, 30);
    setContentView(customSurfaceView);
}

@Override
public void onTargetsLoaded(ImageTracker tracker) {
    dropDownAlert = new DropDownAlert(this);

    dropDownAlert.setText("Escaneja l'imatge per rebre la primera instrucció del
funcionament de la màquina CNC");
    dropDownAlert.setTextWeight(1f);
    dropDownAlert.addImages("powerON.png");
    dropDownAlert.show();

getWindow().addFlags(WindowManager.LayoutParams.FLAG_KEEP_SCREEN_ON);
```

```
}

@Override
public void onErrorLoadingTargets(ImageTracker tracker, WikitudeError error) {
    Log.v(TAG, "Unable to load image tracker. Reason: " + error.getMessage());
}

@Override
public void onImageRecognized(ImageTracker tracker, final ImageTarget target) {
    Log.v(TAG, "Recognized target " + target.getName());
    dropDownAlert.dismiss();
    StrokeredRectangle strokeredRectangle = new
    StrokeredRectangle(StrokeredRectangle.Type.STANDARD);
    glRenderer.setRenderablesForKey(target.getName() + target.getUniqueId(),
    strokeredRectangle, null);
}

@Override
public void onImageTracked(ImageTracker tracker, final ImageTarget target) {
    StrokeredRectangle strokeredRectangle = (StrokeredRectangle)
    glRenderer.getRenderableForKey(target.getName() + target.getUniqueId());

    if (strokeredRectangle != null) {
        strokeredRectangle.viewMatrix = target.getViewMatrix();
        strokeredRectangle.setXScale(target.getTargetScale().x);
        strokeredRectangle.setYScale(target.getTargetScale().y);
    }
}

@Override
public void onImageLost(ImageTracker tracker, final ImageTarget target) {
    Log.v(TAG, "Lost target " + target.getName());
    glRenderer.removeRenderablesForKey(target.getName() + target.getUniqueId());
}

@Override
public void onExtendedTrackingQualityChanged(ImageTracker tracker, final ImageTarget
target, final int oldTrackingQuality, final int newTrackingQuality) {
}
}
```



## 1.4 SegonActivity

```
package com.example.tfgapp1.external;

import android.app.Activity;
import android.os.Bundle;
import android.util.Log;
import android.view.WindowManager;
import android.view.View;
import android.content.Intent;

import com.example.tfgapp1.external.render.CustomSurfaceView;
import com.example.tfgapp1.external.render.Driver;
import com.example.tfgapp1.external.render.DropDownAlert;
import com.example.tfgapp1.external.render.GLRenderer;
import com.example.tfgapp1.external.render.StrokedRectangle;
import com.wititude.NativeStartupConfiguration;
import com.wititude.WititudeSDK;
import com.wititude.common.WititudeError;
import com.wititude.common.camera.CameraSettings;
import com.wititude.common.rendering.RenderExtension;
import com.wititude.rendering.ExternalRendering;
import com.wititude.tracker.ImageTarget;
import com.wititude.tracker.ImageTracker;
import com.wititude.tracker.ImageTrackerListener;
import com.wititude.tracker.TargetCollectionResource;

public class SegundoActivity extends Activity implements ImageTrackerListener,
ExternalRendering {

    private static final String TAG = "SimpleImageTracking2";

    private WititudeSDK wititudeSDK;
    private CustomSurfaceView customSurfaceView;
    private Driver driver;
    private GLRenderer glRenderer;
    private DropDownAlert dropDownAlert;
    private DropDownAlert dropDownAlert2;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);

        wititudeSDK = new WititudeSDK(this);
        NativeStartupConfiguration startupConfiguration = new NativeStartupConfiguration();
        startupConfiguration.setLicenseKey(WaitedSDKSConstants.WAITED_SDKS_KEY);
```

```
startupConfiguration.setCameraPosition(CameraSettings.CameraPosition.BACK);
startupConfiguration.setCameraResolution(CameraSettings.CameraResolution.AUTO);

wikitudeSDK.onCreate(getApplicationContext(), this, startupConfiguration);

final TargetCollectionResource targetCollectionResource =
wikitudeSDK.getTrackerManager().createTargetCollectionResource("file:///android_asset/tracker_2.wtc");
wikitudeSDK.getTrackerManager().createImageTracker(targetCollectionResource, this,
null);
dropDownAlert2 = new DropDownAlert(this);
dropDownAlert2.setText("Pas 2: Ara canviarem de menú, pulsa la tecla shift");
dropDownAlert2.setTextWeight(0.5f);
dropDownAlert2.show();

getWindow().addFlags(WindowManager.LayoutParams.FLAG_KEEP_SCREEN_ON);

}
public void Instru (View view)
{
    Intent Instru2 = new Intent( this, TercerActivity.class);
    startActivity(Instru2);
}

@Override
protected void onResume() {
    super.onResume();
    wikitudeSDK.onResume();
    customSurfaceView.onResume();
    driver.start();
}

@Override
protected void onPause() {
    super.onPause();
    customSurfaceView.onPause();
    driver.stop();
    wikitudeSDK.onPause();
}

@Override
protected void onDestroy() {
    super.onDestroy();
    wikitudeSDK.onDestroy();
}
```

```
@Override
public void onRenderExtensionCreated(final RenderExtension renderExtension) {
    glRenderer = new GLRenderer(renderExtension);

    wikitudeSDK.getCameraManager().setRenderingCorrectedFovChangedListener(glRenderer)
;
    customSurfaceView = new CustomSurfaceView(getApplicationContext(), glRenderer);
    driver = new Driver(customSurfaceView, 30);
    setContentView(customSurfaceView);

}

@Override
public void onTargetsLoaded(ImageTracker tracker) {
    dropDownAlert = new DropDownAlert(this);

    dropDownAlert.setText("Escaneja l'imatge per rebre la segona instrucció del
funcionament de la màquina CNC");
    dropDownAlert.setTextWeight(1f);
    dropDownAlert.addImages("shift_1.png");
    dropDownAlert.show();

    getWindow().addFlags(WindowManager.LayoutParams.FLAG_KEEP_SCREEN_ON);

}

@Override
public void onErrorLoadingTargets(ImageTracker tracker, WikitudeError error) {
    Log.v(TAG, "Unable to load image tracker. Reason: " + error.getMessage());

}

@Override
public void onImageRecognized(ImageTracker tracker, final ImageTarget target) {
    Log.v(TAG, "Recognized target " + target.getName());
    dropDownAlert.dismiss();
    StrokeredRectangle strokedRectangle = new
StrokeredRectangle(StrokeredRectangle.Type.STANDARD);
    glRenderer.setRenderablesForKey(target.getName() +
target.getUniqueId(),strokedRectangle, null);

}

@Override
public void onImageTracked(ImageTracker tracker, final ImageTarget target) {
    StrokeredRectangle strokedRectangle = (StrokeredRectangle)
glRenderer.getRenderableForKey(target.getName() + target.getUniqueId());
```

```
if (strokedRectangle != null) {
    strokedRectangle.viewMatrix = target.getViewMatrix();
    strokedRectangle.setXScale(target.getTargetScale().x);
    strokedRectangle.setYScale(target.getTargetScale().y);
}
```

```
}
```

```
@Override
public void onImageLost(ImageTracker tracker, final ImageTarget target) {
    Log.v(TAG, "Lost target " + target.getName());
    glRenderer.removeRenderablesForKey(target.getName() + target.getUniqueId());
}
```

```
@Override
public void onExtendedTrackingQualityChanged(ImageTracker tracker, final ImageTarget
target, final int oldTrackingQuality, final int newTrackingQuality) {

}

}
```

## 1.5 TercerActivity

```
package com.example.tfgapp1.external;

import android.app.Activity;
import android.os.Bundle;
import android.util.Log;
import android.view.WindowManager;
import android.view.View;
import android.content.Intent;

import com.example.tfgapp1.external.render.CustomSurfaceView;
import com.example.tfgapp1.external.render.Driver;
import com.example.tfgapp1.external.render.DropDownAlert;
import com.example.tfgapp1.external.render.GLRenderer;
import com.example.tfgapp1.external.render.StrokedRectangle;
import com.wikitudo.NativeStartupConfiguration;
```

```
import com.wititude.WititudeSDK;
import com.wititude.common.WititudeError;
import com.wititude.common.camera.CameraSettings;
import com.wititude.common.rendering.RenderExtension;
import com.wititude.rendering.ExternalRendering;
import com.wititude.tracker.ImageTarget;
import com.wititude.tracker.ImageTracker;
import com.wititude.tracker.ImageTrackerListener;
import com.wititude.tracker.TargetCollectionResource;

public class TercerActivity extends Activity implements ImageTrackerListener,
ExternalRendering {

    private static final String TAG = "SimpleImageTracking3";

    private WititudeSDK wititudeSDK;
    private CustomSurfaceView customSurfaceView;
    private Driver driver;
    private GLRenderer glRenderer;
    private DropDownAlert dropDownAlert;
    private DropDownAlert dropDownAlert2;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);

        wititudeSDK = new WititudeSDK(this);
        NativeStartupConfiguration startupConfiguration = new NativeStartupConfiguration();
        startupConfiguration.setLicenseKey(WaitedSDKSConstants.WAITED_SDKS_KEY);
        startupConfiguration.setCameraPosition(CameraSettings.CameraPosition.BACK);
        startupConfiguration.setCameraResolution(CameraSettings.CameraResolution.AUTO);

        wititudeSDK.onCreate(getApplicationContext(), this, startupConfiguration);

        final TargetCollectionResource targetCollectionResource =
        wititudeSDK.getTrackerManager().createTargetCollectionResource("file:///android_asset/tracker_3.wtc");
        wititudeSDK.getTrackerManager().createImageTracker(targetCollectionResource, this,
        null);
        dropDownAlert2 = new DropDownAlert(this);
        dropDownAlert2.setText("Pas 3: A continuació prem la tecla ESC, torna a apretar el
        shift + esc, si s'ha canviat el menú fés click en següent ");
        dropDownAlert2.setTextWeight(0.5f);
        dropDownAlert2.show();

        getWindow().addFlags(WindowManager.LayoutParams.FLAG_KEEP_SCREEN_ON);
    }
}
```

```
}
public void Instru (View view)
{
    Intent Instru2 = new Intent( this, CuartoActivity.class);
    startActivity(Instru2);
}

@Override
protected void onResume() {
    super.onResume();
    wiktitudeSDK.onResume();
    customSurfaceView.onResume();
    driver.start();
}

@Override
protected void onPause() {
    super.onPause();
    customSurfaceView.onPause();
    driver.stop();
    wiktitudeSDK.onPause();
}

@Override
protected void onDestroy() {
    super.onDestroy();
    wiktitudeSDK.onDestroy();
}

@Override
public void onRenderExtensionCreated(final RenderExtension renderExtension) {
    glRenderer = new GLRenderer(renderExtension);

wiktitudeSDK.getCameraManager().setRenderingCorrectedFovChangedListener(glRenderer)
;
    customSurfaceView = new CustomSurfaceView(getApplicationContext(), glRenderer);
    driver = new Driver(customSurfaceView, 30);
    setContentView(customSurfaceView);
}

@Override
public void onTargetsLoaded(ImageTracker tracker) {
    dropDownAlert = new DropDownAlert(this);
```

```
        dropDownAlert.setText("Escaneja l'imatge per rebre la tercera instrucció del
funcionament de la màquina CNC");
        dropDownAlert.setTextWeight(1f);
        dropDownAlert.addImages("ESC_1.png");
        dropDownAlert.show();

        getWindow().addFlags(WindowManager.LayoutParams.FLAG_KEEP_SCREEN_ON);

    }

    @Override
    public void onErrorLoadingTargets(ImageTracker tracker, WikitudeError error) {
        Log.v(TAG, "Unable to load image tracker. Reason: " + error.getMessage());
    }

    @Override
    public void onImageRecognized(ImageTracker tracker, final ImageTarget target) {
        Log.v(TAG, "Recognized target " + target.getName());
        dropDownAlert.dismiss();
        StrokeredRectangle strokeredRectangle = new
        StrokeredRectangle(StrokeredRectangle.Type.STANDARD);
        glRenderer.setRenderablesForKey(target.getName() +
        target.getUniqueId(),strokeredRectangle, null);
    }

    @Override
    public void onImageTracked(ImageTracker tracker, final ImageTarget target) {
        StrokeredRectangle strokeredRectangle = (StrokeredRectangle)
        glRenderer.getRenderableForKey(target.getName() + target.getUniqueId());

        if (strokeredRectangle != null) {
            strokeredRectangle.viewMatrix = target.getViewMatrix();
            strokeredRectangle.setXScale(target.getTargetScale().x);
            strokeredRectangle.setYScale(target.getTargetScale().y);
        }
    }

    @Override
    public void onImageLost(ImageTracker tracker, final ImageTarget target) {
        Log.v(TAG, "Lost target " + target.getName());
    }
}
```

```
        glRenderer.removeRenderablesForKey(target.getName() + target.getUniqueId());
    }

    @Override
    public void onExtendedTrackingQualityChanged(ImageTracker tracker, final ImageTarget
target, final int oldTrackingQuality, final int newTrackingQuality) {

    }

}
```

## 1.6 QuartActivity

```
package com.example.tfgapp1.external;

import android.app.Activity;
import android.os.Bundle;
import android.util.Log;
import android.view.WindowManager;
import android.view.View;
import android.content.Intent;

import com.example.tfgapp1.external.render.CustomSurfaceView;
import com.example.tfgapp1.external.render.Driver;
import com.example.tfgapp1.external.render.DropDownAlert;
import com.example.tfgapp1.external.render.GLRenderer;
import com.example.tfgapp1.external.render.StrokedRectangle;
import com.wikitude.NativeStartupConfiguration;
import com.wikitude.WikitudeSDK;
import com.wikitude.common.WikitudeError;
import com.wikitude.common.camera.CameraSettings;
import com.wikitude.common.rendering.RenderExtension;
import com.wikitude.rendering.ExternalRendering;
import com.wikitude.tracker.ImageTarget;
import com.wikitude.tracker.ImageTracker;
import com.wikitude.tracker.ImageTrackerListener;
import com.wikitude.tracker.TargetCollectionResource;

public class CuartoActivity extends Activity implements ImageTrackerListener,
ExternalRendering {

    private static final String TAG = "SimpleImageTracking4";

    private WikitudeSDK wikitudeSDK;
    private CustomSurfaceView customSurfaceView;
    private Driver driver;
```



```
private GLRenderer glRenderer;
private DropDownAlert dropDownAlert;
private DropDownAlert dropDownAlert2;

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);

    wiktudeSDK = new WiktudeSDK(this);
    NativeStartupConfiguration startupConfiguration = new NativeStartupConfiguration();
    startupConfiguration.setLicenseKey(WaitedSDKSConstants.WAITED_SDKS_KEY);
    startupConfiguration.setCameraPosition(CameraSettings.CameraPosition.BACK);
    startupConfiguration.setCameraResolution(CameraSettings.CameraResolution.AUTO);

    wiktudeSDK.onCreate(getApplicationContext(), this, startupConfiguration);

    final TargetCollectionResource targetCollectionResource =
    wiktudeSDK.getTrackerManager().createTargetCollectionResource("file:///android_asset/tracker_4.wtc");
    wiktudeSDK.getTrackerManager().createImageTracker(targetCollectionResource, this,
    null);
    dropDownAlert2 = new DropDownAlert(this);
    dropDownAlert2.setText("Pas 4: Ara procedim a fer un canvi d'eina, prem el botó
    hexagonal, si canvia l'eina del torn fés click en següent ");
    dropDownAlert2.setTextWeight(0.5f);
    dropDownAlert2.show();

    getWindow().addFlags(WindowManager.LayoutParams.FLAG_KEEP_SCREEN_ON);
}
public void Instru (View view)
{
    Intent Instru2 = new Intent( this, CuartoActivity.class);
    startActivity(Instru2);
}

@Override
protected void onResume() {
    super.onResume();
    wiktudeSDK.onResume();
    customSurfaceView.onResume();
    driver.start();
}
```

```
@Override
protected void onPause() {
    super.onPause();
    customSurfaceView.onPause();
    driver.stop();
    wiktitudeSDK.onPause();
}

@Override
protected void onDestroy() {
    super.onDestroy();
    wiktitudeSDK.onDestroy();
}

@Override
public void onRenderExtensionCreated(final RenderExtension renderExtension) {
    glRenderer = new GLRenderer(renderExtension);

wiktitudeSDK.getCameraManager().setRenderingCorrectedFovChangedListener(glRenderer)
;
    customSurfaceView = new CustomSurfaceView(getApplicationContext(), glRenderer);
    driver = new Driver(customSurfaceView, 30);
    setContentView(customSurfaceView);
}

@Override
public void onTargetsLoaded(ImageTracker tracker) {
    dropDownAlert = new DropDownAlert(this);

    dropDownAlert.setText("Escaneja l'imatge per rebre la quarta instrucció del
funcionament de la màquina CNC");
    dropDownAlert.setTextWeight(1f);
    dropDownAlert.addImages("midbutton.png");
    dropDownAlert.show();

    getWindow().addFlags(WindowManager.LayoutParams.FLAG_KEEP_SCREEN_ON);
}

@Override
public void onErrorLoadingTargets(ImageTracker tracker, WikitudeError error) {
    Log.v(TAG, "Unable to load image tracker. Reason: " + error.getMessage());
}

@Override
public void onImageRecognized(ImageTracker tracker, final ImageTarget target) {
```

```
        Log.v(TAG, "Recognized target " + target.getName());
        dropdownAlert.dismiss();
        StrokeredRectangle strokedRectangle = new
StrokeredRectangle(StrokeredRectangle.Type.STANDARD);
        glRenderer.setRenderablesForKey(target.getName() +
target.getUniqueld(),strokedRectangle, null);

    }

    @Override
    public void onImageTracked(ImageTracker tracker, final ImageTarget target) {
        StrokeredRectangle strokedRectangle = (StrokeredRectangle)
glRenderer.getRenderableForKey(target.getName() + target.getUniqueld());

        if (strokedRectangle != null) {
            strokedRectangle.viewMatrix = target.getViewMatrix();
            strokedRectangle.setXScale(target.getTargetScale().x);
            strokedRectangle.setYScale(target.getTargetScale().y);

        }

    }

    @Override
    public void onImageLost(ImageTracker tracker, final ImageTarget target) {
        Log.v(TAG, "Lost target " + target.getName());
        glRenderer.removeRenderablesForKey(target.getName() + target.getUniqueld());
    }

    @Override
    public void onExtendedTrackingQualityChanged(ImageTracker tracker, final ImageTarget
target, final int oldTrackingQuality, final int newTrackingQuality) {

    }

}
```