



# NPI- PRÁCTICA 3

## TUTORIAL - GESTOS

Javier Escobar Cerezo  
Julio Rodríguez Martínez

# GESTOS PERSONALIZADOS

- Surge por la necesidad de simplificación de interfaces en dispositivos android
- A partir de la SDK 1.6 (API 4) se incluye la librería Gesture. (android.gesture). Esta posibilita:
  - Dibujar gestos nuevos
  - Almacenar gestos
  - Cargar gestos
  - Reconocer gestos



# GESTURE BUILDER / TOOL



- El SDK de Android suministra una aplicación llamada *Gesture Builder / Gesture Tool* esta permite crear nuestros gestos personalizados  
<https://play.google.com/store/apps/details?id=com.davemacs327.gesture.tool&hl=es>
- Los gestos creados se guardan en fichero “gesture.txt” dentro de la tarjeta de memoria de nuestro dispositivo.
- Este archivo se incluirá dentro de nuestro nuevo proyecto haciendo una instancia a la librería *GestureLibrary* . De esta manera se podrá usar para buscar coincidencias con nuestros gestos personalizados.



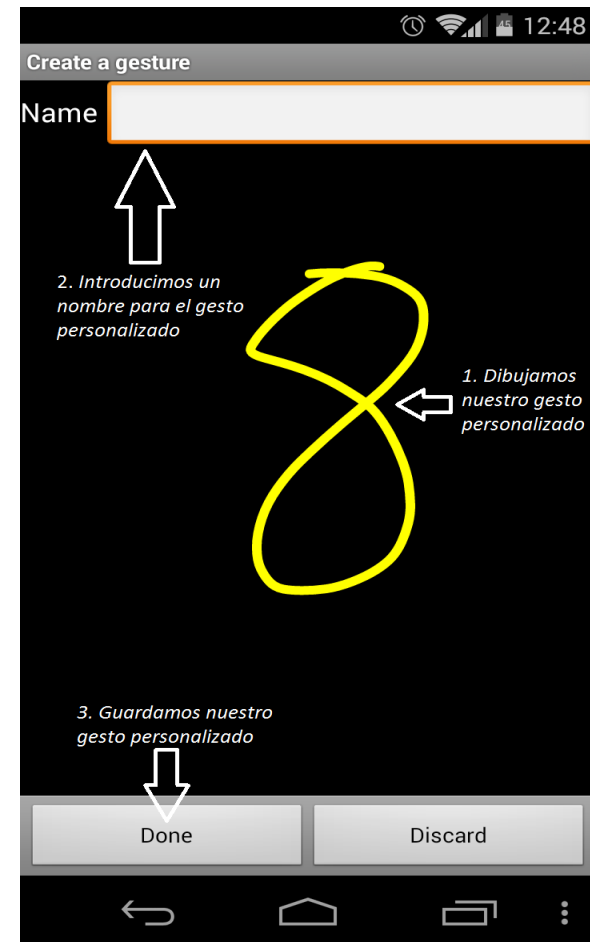
# CREACIÓN DE APK DE PRUEBA

- Primer paso: Crear un aplicación vacía con eclipse. *File->New->Project->Android Application Project.*
  - Minimun Requiered SDK: API 7: Android 2.1 (Eclair)
- Segundo Paso: Creación e incorporación de gestos al proyecto
  - Descargar *Gesture Tool*  
<https://play.google.com/store/apps/details?id=com.davemac327.gesture.tool&hl=es>



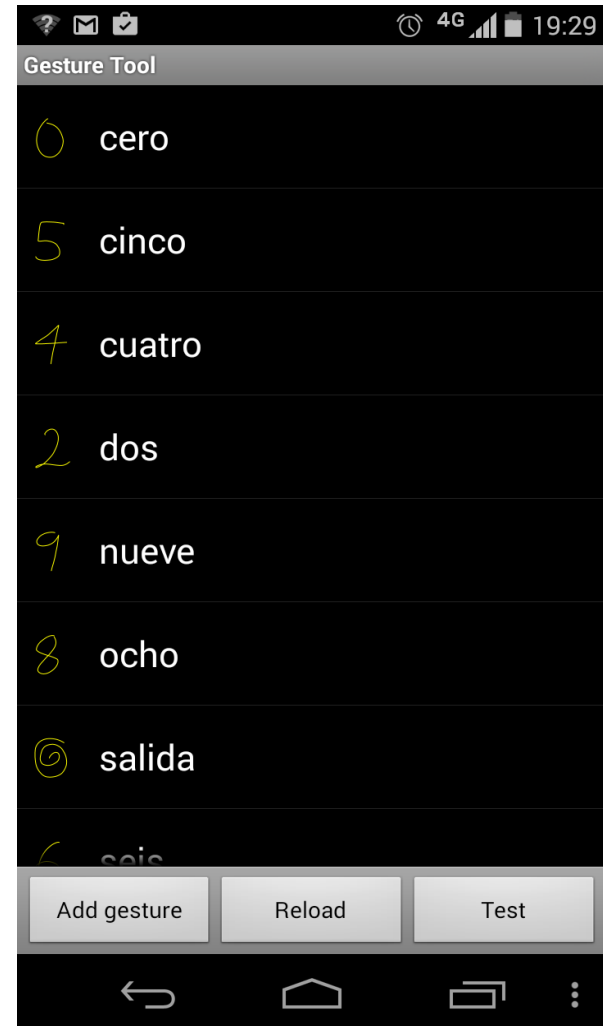
## CREACIÓN DE APK DE PRUEBA (II)

- Abrir y pulsar *add gesture* para añadir
- Arrastrar el dedo sobre la pantalla para definir la forma
- Nombrar forma y pulsar *done*.



# CREACIÓN DE APK DE PRUEBA (III)

- Repite el proceso hasta tener todos los gestos que quieras.
  - En nuestro caso hemos guardado un gesto por cada número 0-9 y un gesto para la salida
- Con el botón *test* puedes probar los gestos creados
- En *sdcard/gestures* se creará un fichero con nuestros gestos personalizados. “gesture.txt”
- Este fichero lo pasamos al proyecto guardándolo en la ruta:
  - */res/raw*



# CREACIÓN DE APK DE PRUEBA (IV)

- Tercer paso: Inclusión de librerías en el proyecto.  
(*Main.Activity.java*)

```
import android.gesture.Gesture;  
import android.gesture.GestureLibraries;  
import android.gesture.GestureLibrary;  
import android.gesture.GestureOverlayView;  
import android.gesture.GestureOverlayView.OnGesturePerformedListener;  
import android.gesture.Prediction;
```

- *Gesuture*: todas las librerías asociadas con gestos
- *GestureLibraries*: Lectura y escritura de gestos y su manejo
- *GestureLibrary*: Cargar, crear gestos, determinar gesto introducido, secuencia de un gesto, etc.
- *GestureOverlayView*: método OnGesturePermormedListener
- *Prediction*: métodos probalisticos para la detección de gestos, patrones, orientaciones, etc.
- Más información sobre las librerías:  
<http://developer.android.com/reference/android/gesture/package-summary.html>



# CREACIÓN DE APK DE PRUEBA (V)

- Cuarto paso: Definición de constructores y métodos en la aplicación.
  - Res/layout/activity\_main\_activity2.xml

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    >

    <TextView
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:text="@string/info"/>

    <android.gesture.GestureOverlayView
        android:id="@+id/gestures"
        android:layout_width="fill_parent"
        android:layout_height="fill_parent"
        android:gestureStrokeType="multiple"
        android:fadeOffset="800"/>
</LinearLayout>
```

- Tenemos un *layout* formado por un *LinearLayout* que contiene: un *TextView* con un título y un *GestureOverlayView* para la salida del programa. *GestureStrokeType* permite varios trazos y *fadeoffset* es el tiempo entre gestos múltiples.





# CREACIÓN DE APK DE PRUEBA (VI)

- Modificaciones en main src/MainActivity2.java

```
public class MainActivity2 extends Activity implements OnGesturePerformedListener
{
    ...
    private GestureLibrary gestureLib;
    @Override
    protected void onCreate(Bundle savedInstanceState)
    {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main_activity2);

        /*Detección de gestos (solo se abre al principio de la aplicación). */

        GestureOverlayView gestureOverlayView = new GestureOverlayView(this);
        View inflate = getLayoutInflater().inflate(R.layout.activity_main_activity2, null);
        gestureOverlayView.addView(inflate);
        gestureOverlayView.addOnGesturePerformedListener(this);
        gestureLib = GestureLibraries.fromRawResource(this, R.raw.gestures);

        if (!gestureLib.load())
        {
            finish();
        }
        setContentView(gestureOverlayView);
    }
}
```



# CREACIÓN DE APK DE PRUEBA (VI)

```
/* Métodos para tratamiento de gestos. */
@Override
public void onGesturePerformed(GestureOverlayView overlay, Gesture gesture)
{
    ArrayList<Prediction> predictions = gestureLib.recognize(gesture);
    for (Prediction prediction : predictions)
    {
        //Otra opción es directamente coger el primer elemento que supere la predicción de 1.0
        if (prediction.score > 3.0)
        {
            if(prediction.name.equals("salida"))
            {
                Toast.makeText(this, "Hasta pronto", Toast.LENGTH_SHORT).show();
                super.finish();
            }
            else
            {
                if(cont%2==0 && cont!=0)
                {
                    Toast.makeText(this, prediction.name, Toast.LENGTH_SHORT).show();
                    num1=quenúmero(prediction.name);
                    int valor=num1+num;
                    Toast.makeText(this, num+" + "+num1+" = "+valor, Toast.LENGTH_SHORT).show();
                }
                else
                {
                    Toast.makeText(this, prediction.name, Toast.LENGTH_SHORT).show();
                    num=quenúmero(prediction.name);
                }
                cont++;
            }
        }
    }
}

public int quenúmero(String número)
```



# CREACIÓN DE APK DE PRUEBA (VI)

- Quinto paso. Prueba de la aplicación



- La aplicación reconoce números del 0-9. Si pones dos números consecutivos los suma automática y da el resultado. Si pintas un espiral la aplicación termina.
- Enlace GITHUB  
<https://github.com/Jick9536/NPI-P3>



# PROBLEMAS ENCONTRADOS

- La predicción por defecto 1.0 es muy baja hay que ajustarla bien para cada tipo de dispositivo si no se quiere que se reconozcan varios gestos simultáneamente.
- Los gestos tienen que hacerse en un solo trazo. Aunque *gesture tool* que permite hacer gestos múltiples a la hora de hacer el ejemplo sólo reconoce un solo trazo. Este problema debería poder solucionarse añadiendo *android:gestureStrokeType="multiple"* y *android:fadeOffset="800"/* en el xml



## EJEMPLOS DE USO DE LIBRERÍA DE GESTURE

<http://www.apksforandroid.com/back-button-gesture-launcher-1-5-apk/>

<http://www.1mobile.es/finger-gesture-launcher-38666.html>

<https://gitorious.org/g2l-gesture-launcher/g2l-gesture-launcher/source/8df977fa18e8ac761a4c0a4da3be41b3270d5bb1>

Gesture Search <http://www.youtube.com/watch?v=mW9r7hXi8mk>

Gesture Launcher <https://gitorious.org/g2l-gesture-launcher/g2l-gesture-launcher/source/8df977fa18e8ac761a4c0a4da3be41b3270d5bb1> <http://www.youtube.com/watch?v=-lZazZPbaP8>

Move'n Launch Lite <http://www.youtube.com/watch?v=V-jUvLj3J9E>

Sign <https://www.youtube.com/watch?v=2QsyXtqVAEw#action=share>

## BIBLIOGRAFÍA

[http://www.techotopia.com/index.php/Implementing Android Custom Gesture Recognition with Android Studio](http://www.techotopia.com/index.php/Implementing_Android_Custom_Gesture_Recognition_with_Android_Studio)

<http://developer.android.com/training/gestures/detector.html>

<http://developer.android.com/training/gestures/movement.html>

<http://developer.android.com/training/gestures/multi.html>

<http://developer.android.com/training/gestures/viewgroup.html>

<http://developer.android.com/reference/android/gesture/package-summary.html>

<https://nuevos-paradigmas-de-interaccion.wikispaces.com/Detecci%C3%B3n+de+patrones+en+Android+-+Gesture+Builder>

[http://nuevos-paradigmas-de-interaccion.wikispaces.com/file/view/PresentacionP3\\_Android\\_3Sensores.pdf/535961100/PresentacionP3\\_Android\\_3Sensores.pdf](http://nuevos-paradigmas-de-interaccion.wikispaces.com/file/view/PresentacionP3_Android_3Sensores.pdf/535961100/PresentacionP3_Android_3Sensores.pdf)

<https://androidresearch.wordpress.com/tag/gesture-builder/>

<http://www.androidcurso.com/index.php/tutoriales-android/36-unidad-5-entradas-en-android-teclado-pantalla-tactil-y-sensores/153-gestures>

<http://secretos-android.blogspot.com.es/2011/07/los-gestos-en-android.html>