



Instituto Tecnológico del Valle de Oaxaca.

INGENIERÍA EN TIC'S.

Aplicaciones Móviles II

Bitácora sensor Giroscopio

Docente: Ambrosio Cardoso Jiménez.

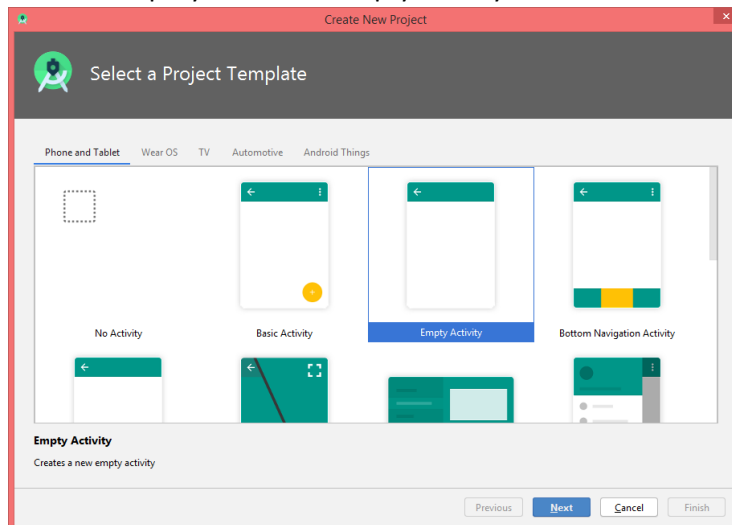
Alumna: Araceli Ibañez Cruz.



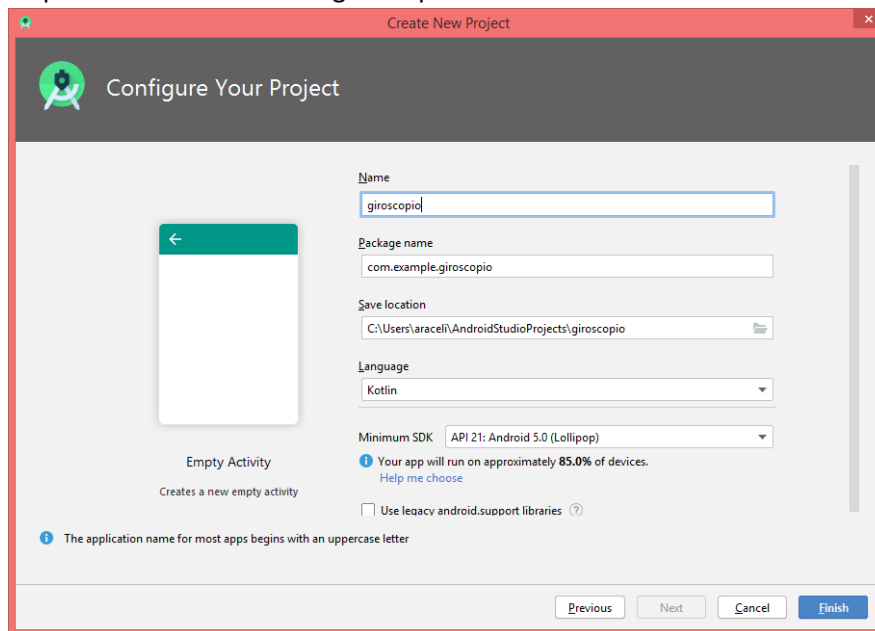
Ex Hacienda de Nazareno, Xoxocotlán, Oaxaca.

2 de marzo de 2020.

1. Creamos el proyecto en un empty activity



2. Le ponemos el nombre de giroscopio3.



3. Creamos un las siguientes carpetas:

Models:

- MiSensor.kt

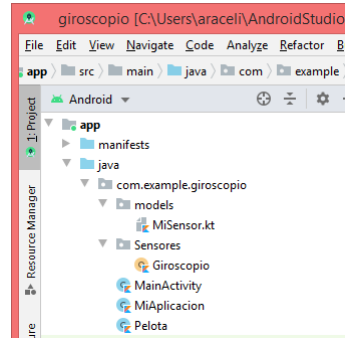
Sensores

- Giroscopio

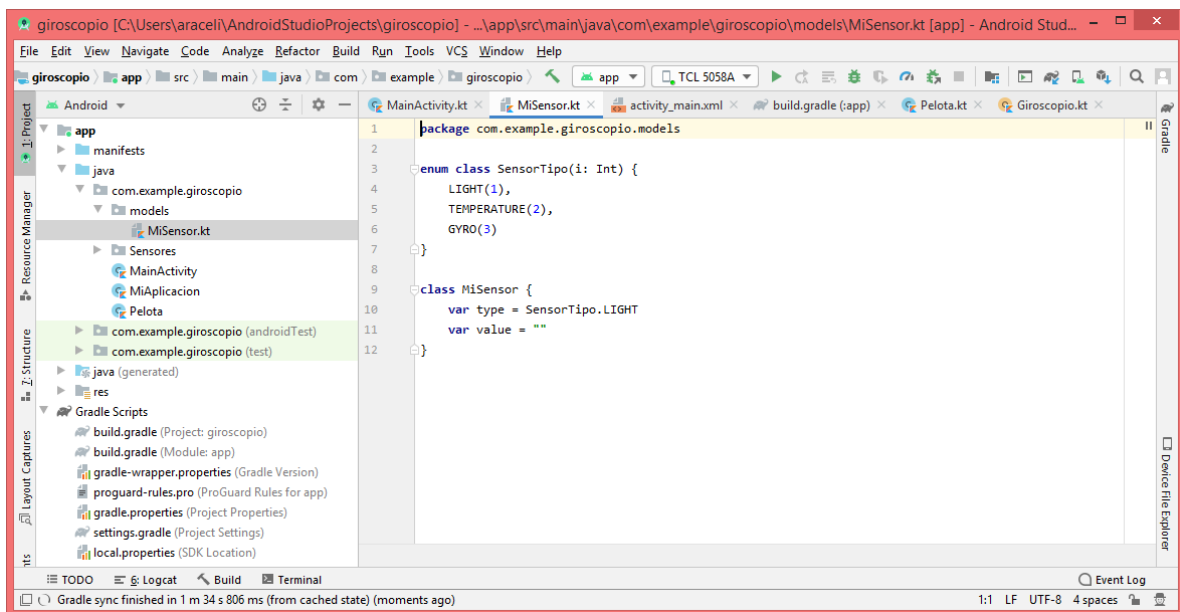
MainActivity

MiAplicacion

Pelota



4. E la clase de miSensor.kt Agregamos el siguiente código



5. El la clase de giroscopio agregamos
`package com.example.giroscopio.Sensores`

```
import android.app.Service
import android.hardware.Sensor
import android.hardware.SensorEvent
import android.hardware.SensorEventListener
import android.hardware.SensorManager
import android.os.Build
import android.os.Handler
import android.os.HandlerThread
import android.util.Log
import androidx.annotation.RequiresApi
import com.example.giroscopio.MiAplicacion
```

```

import com.example.giroscopio.models.MiSensor
import com.example.giroscopio.models.SensorTipo

object Giroscopio :
    HandlerThread("Giroscopio"), SensorEventListener {
        var alto:Int?=null
        var ancho:Int?=null
        var ejeX:Float=0F
        var ejeY:Float=0F
        var ejeZ:Float=0F
        private val TAG : String = "Giroscopio"
        private var handler: Handler? = null
        private var sensorManager : SensorManager?= null
        private var sensor : Sensor?= null
        private var sensorExists = false
        private var sensorThread: HandlerThread? = null
        private var sensorHandler: Handler? = null

        init{

            sensorManager =
(MiAplicacion.getApplicationContext().getSystemService(Service.SENSOR_SER
VICE)) as SensorManager
            sensor = sensorManager!!.getDefaultSensor(Sensor.TYPE_GYROSCOPE)

            // Check sensor exists
            if (sensor != null) {
                sensorExists = true
            } else {
                sensorExists = false
            }
            this.alto=500
            this.ancho=800
        }

        fun startSensor(){
            sensorThread = HandlerThread(TAG, Thread.NORM_PRIORITY)
            sensorThread!!.start()
            sensorHandler = Handler(sensorThread!!.getLooper()) //Blocks
until looper is prepared, which is fairly quick
            sensorManager!!.registerListener(this,
                sensor, SensorManager.SENSOR_DELAY_NORMAL,
                sensorHandler
            )
        }
        fun Pantalla(ancho:Int, alto:Int){
            this.ancho=ancho
            this.alto=alto
        }
    }

```

6. En la clase principal MainActivity agregamos lo siguiente

```
package com.example.giroscopio

import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.graphics.Bitmap
import android.graphics.Canvas
import android.os.Handler
import android.os.Looper
import android.os.Message
import android.view.View
import com.example.giroscopio.Sensores.Giroscopio
import com.example.giroscopio.models.MiSensor
import com.example.giroscopio.models.SensorTipo

import kotlinx.android.synthetic.main.activity_main.*

class MainActivity : AppCompatActivity() {

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)

        Giroscopio.setHandler(handler)
        //Giroscopio.Pantalla(ancho,alto)
        btnIniciar.setOnClickListener(object : View.OnClickListener{
            override fun onClick(v: View?) {

                if (!Giroscopio.sensorExists()){
                    txtvSensorGiro.text = "No hay sensor de giroscopio"
                }
                else {
                    Giroscopio.startSensor()
                }

            })
        btnDetener.setOnClickListener(object : View.OnClickListener{
            override fun onClick(v: View?) {

                Giroscopio.stopSensor()
                txtvSensorGiro.text = ""
                ImageViewPelota.setImageBitmap(null)
            })
    }
    //---Control de mensajes
    val handler: Handler = object : Handler(Looper.getMainLooper()) {
        override fun handleMessage(inputMessage: Message) {

            val sensorEvent = inputMessage.obj as MiSensor

            //--- Giroscopio
```

```

        if (sensorEvent.type == SensorTipo.GYRO){

            var str= sensorEvent.value
            var delimiter=","
            var parts= str.split(delimiter)
            txtvSensorGiro.text = "X: ${parts[0]} Y: ${parts[1]} Z:
            ${parts[2]} \n W:${parts[3]} H:${parts[4]} \n Valores de X y Y Adaptados"

            val v: View =Pelota(this@MainActivity,
            parts[0].toFloat(),parts[1].toFloat(), parts[2].toFloat())
            val bitmap =
                Bitmap.createBitmap(800 /*width*/, 500 /*height*/,
                Bitmap.Config.ARGB_8888)
            var canvas= Canvas(bitmap)
            v.draw(canvas)
            ImageViewPelota.setImageBitmap(bitmap)

        }
    }
}

```

7. En la clase de MiAplicación

```
package com.example.giroscopio
```

```
import android.app.Application
import android.content.Context
```

```
class MiAplicacion : Application() {

    init {
        instancia = this
    }
    companion object {
        private var instancia: MiAplicacion?=null

        fun getApplicationContext() : Context {
            return instancia!!.applicationContext
        }
    }
}

```

8. En la clase pelota agregamos el siguiente código

```
package com.example.giroscopio

import android.app.Service
import android.content.Context
import android.graphics.Canvas
import android.graphics.Color
import android.graphics.Paint
import android.view.Display
import android.view.View

```

```
class Pelota(context: Context?, var ejeX: Float, var ejeY: Float, var
ejeZ: Float): View(context) {
```

```
    override fun onDraw(pelota: Canvas){
        var pincel= Paint()
        pincel.setColor(Color.RED)
        pelota.drawCircle(ejeX, ejeY, 40F, pincel)
    }
```

```
}
```

9. En el activity_main ponemos lo siguiente

```
<?xml version="1.0" encoding="utf-8"?>
<ScrollView xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity"
    >
    <androidx.constraintlayout.widget.ConstraintLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        tools:context=".MainActivity">

        <Button
            android:id="@+id/btnIniciar"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginTop="28dp"
            android:text="@string/Iniciar"
            app:layout_constraintEnd_toStartOf="@+id/btnDetener"
            app:layout_constraintHorizontal_bias="0.4"
            app:layout_constraintStart_toStartOf="parent"
            app:layout_constraintTop_toTopOf="parent" />

        <Button
            android:id="@+id/btnDetener"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginEnd="80dp"
            android:text="@string/Detener"
            app:layout_constraintBottom_toTopOf="@+id/textView"
            app:layout_constraintEnd_toEndOf="parent"
            app:layout_constraintTop_toTopOf="@+id/btnIniciar"
            app:layout_constraintVertical_bias="0.0" />
```

```

<TextView
    android:id="@+id/textView7"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="32dp"
    android:text="@string/SensorGiro"
    android:textAppearance="@style/TextAppearance.AppCompat.Body2"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.214"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/termometro" />

```

```

<TextView
    android:id="@+id/txtvSensorGiro"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="32dp"
    android:text="@string/GiroXYZ"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.0"
    app:layout_constraintStart_toStartOf="@+id/textView7"
    app:layout_constraintTop_toBottomOf="@+id/textView7" />

```

```

<ImageView
    android:id="@+id/ImageViewPelota"
    android:layout_width="800px"
    android:layout_height="500px"
    android:layout_marginTop="32dp"
    android:layout_marginBottom="24dp"
    android:background="#03A9F4"
    android:padding="1dp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/txtvSensorGiro" />

```

```

</androidx.constraintlayout.widget.ConstraintLayout>
</ScrollView>

```

10. En la carpeta de values en

- **String.xml** agregamos

```

<resources>
    <string name="app_name">Giroscopio</string>
    <string name="Iniciar">Iniciar</string>

```



```

<string name="Detener">Detener</string>
  <string name="SensorGiro">Sensor giroscopio</string>
  <string name="GiroXYZ">XYZ</string>
  <string name="applay">Aplicar:</string>
</resources>

```

- **En dims.xml**

```

<resources>
  <!-- Default screen margins, per the Android Design guidelines. -->
  <dimen name="activity_horizontal_margin">16dp</dimen>
  <dimen name="activity_vertical_margin">16dp</dimen>
</resources>

```

11. Ejecución de la aplicación

