



DESARROLLO DE APLICACIONES MÓVILES AVANZADAS



“PROGRAMACIÓN ORIENTADA A OBJETOS”

Resultados de Aprendizaje:

- Comprender los elementos necesarios para iniciarse en el desarrollo de aplicaciones basadas en Android.
- Desarrollar pequeños proyectos haciendo uso de la programación orientada a objetos con Kotlin en Android.

INDICE

INTRODUCCIÓN	3
CREANDO NUESTRO PROYECTO	3
PRÁCTICA A REALIZAR	15
RUBRICA DE EVALUACIÓN	16

INTRODUCCIÓN

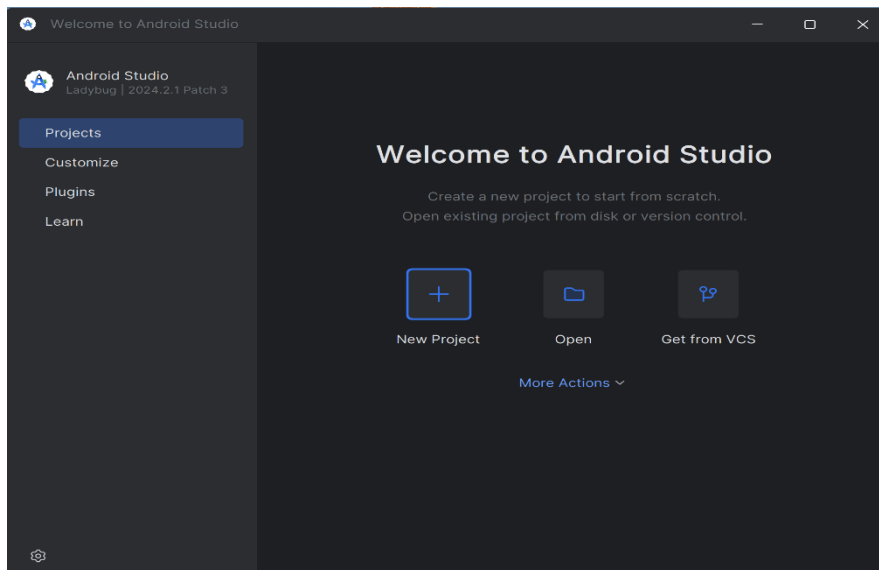
Para desarrollar esta práctica usted deberá tener los siguientes paquetes de software:

- Android Studio Versión 4 o superior
- Equipo Móvil con Android (Puede ser Tablet o Smartphone) con **modo desarrollador habilitado**.

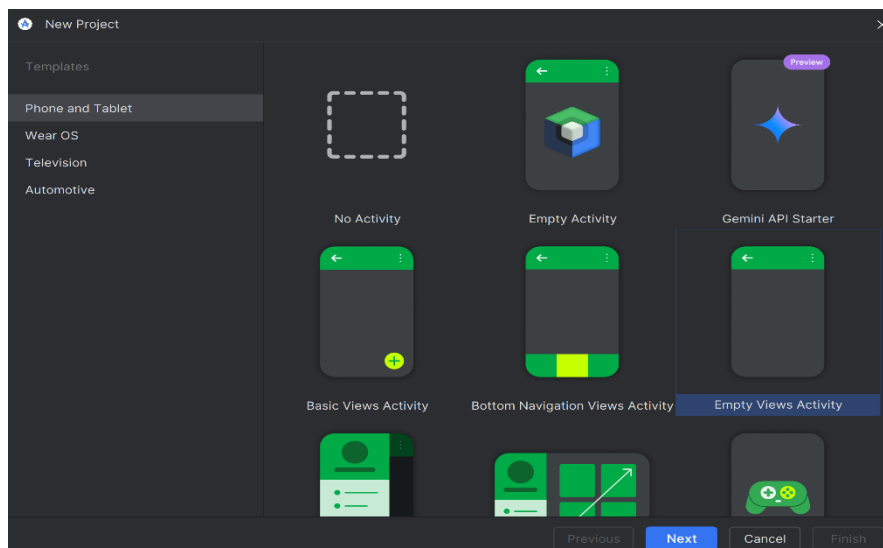
(Ver <https://www.youtube.com/watch?v=wLJS8CKo95o>)

CREANDO NUESTRO PROYECTO

1. Iniciamos nuestro Android Studio.

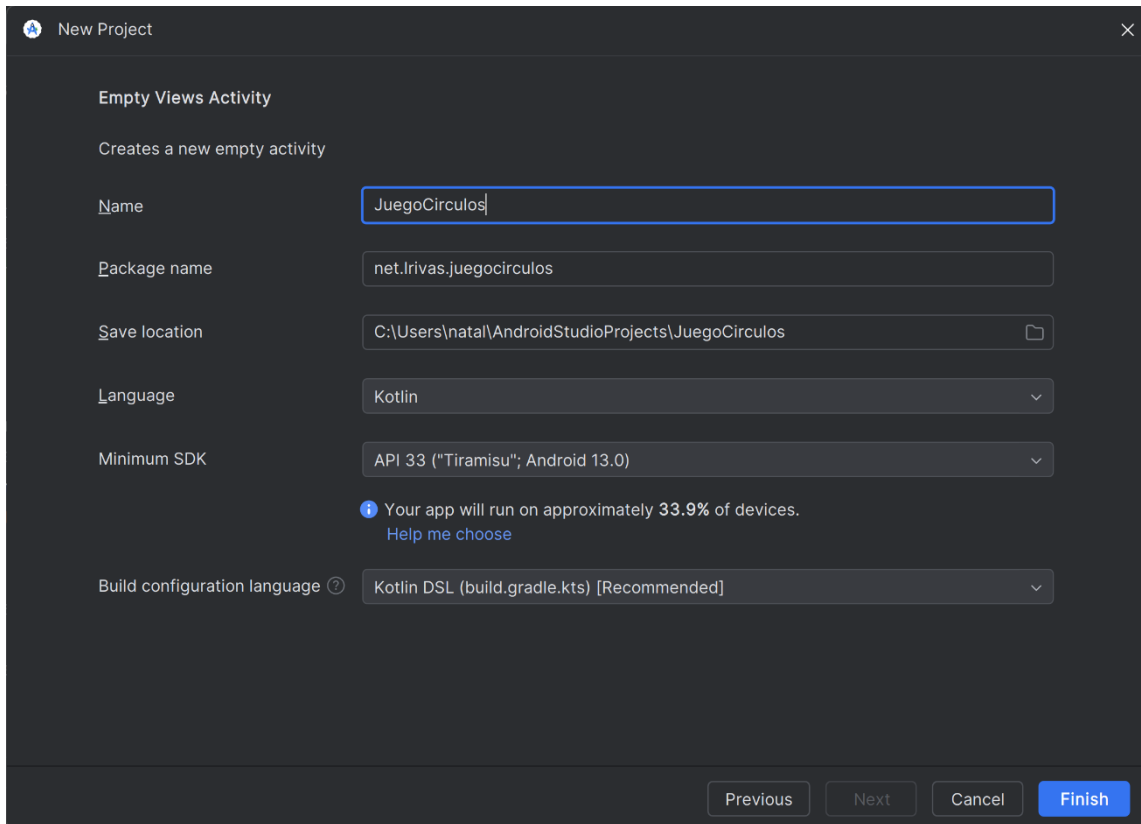


2. Elegimos la plantilla **Empty Views Activity**.



3. Luego establezca las siguientes propiedades al proyecto:
 - a. Nombre: **JuegoCirculos**

- b. Package Name: **net.lrivas.juegocirculos**
- c. Lenguaje: **Kotlin**
- d. Minimum Api: **33**



4. Una vez que se genere el proyecto, de clic sobre **activity_main.xml**. Y copie y pegue el siguiente contenido:

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

    <TextView
        android:id="@+id/headerText"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:background="@color/black"
        android:padding="16dp"
        android:text="Juego de Circulos"
        android:textAlignment="center"
        android:textColor="@color/game_text"
        android:textSize="24sp"
        android:textStyle="bold"
        app:layout_constraintTop_toTopOf="parent" />

    <Button
```

```

        android:id="@+id/ButtonIniciar"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Iniciar Juego"
        android:textSize="18sp"
        app:layout_constraintBottom_toTopOf="@id/footerText"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@id/headerText" />

<TextView
    android:id="@+id/footerText"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:background="@color/black"
    android:padding="8dp"
    android:text="© 2025 Todos los derechos reservados"
    android:textAlignment="center"
    android:textColor="@color/footer_text"
    android:textSize="14sp"
    app:layout_constraintBottom_toBottomOf="parent" />

</androidx.constraintlayout.widget.ConstraintLayout>

```

5. Ahora cree 2 archivos en el drawable, el primero se debe llamar circle_green.xml y pegue el siguiente código.

```

<?xml version="1.0" encoding="utf-8"?>
<shape xmlns:android="http://schemas.android.com/apk/res/android"
    android:shape="oval">
    <solid android:color="#00FF00" />
</shape>

```

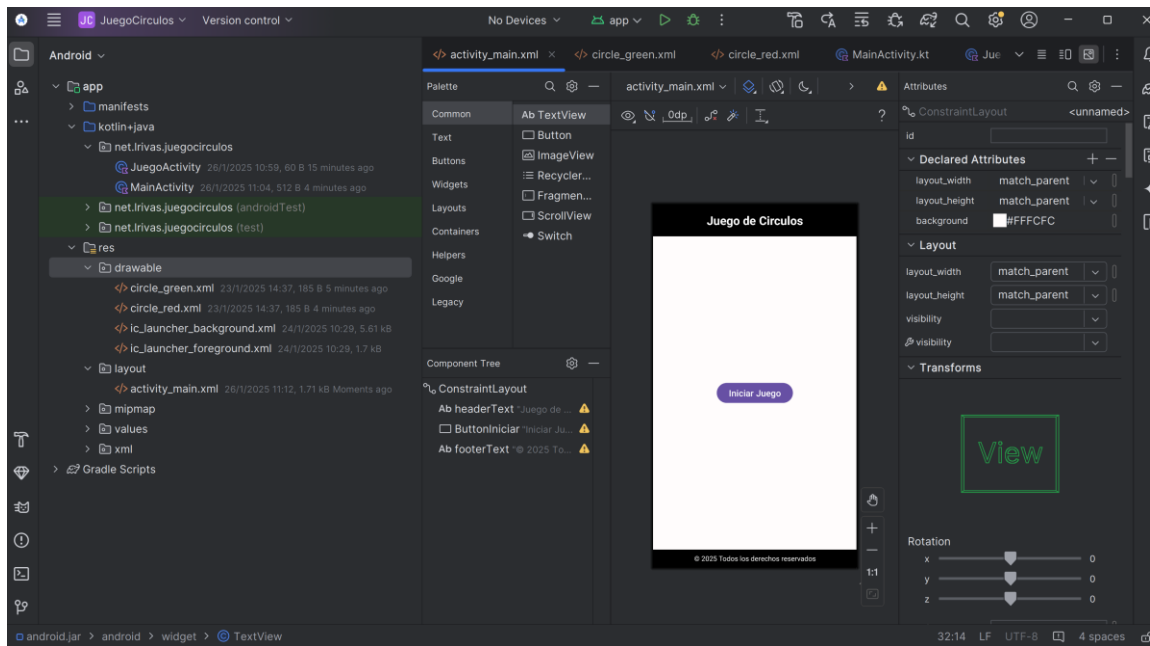
El siguiente archivo nombrelo circle_red.xml y coloque el siguiente código.

```

<?xml version="1.0" encoding="utf-8"?>
<shape xmlns:android="http://schemas.android.com/apk/res/android"
    android:shape="oval">
    <solid android:color="#FF0000" />
</shape>

```

6. Deberá mostrarse una interfaz como la que se ve a continuación al momento de cambiar de la vista de **Code** a vista de **Design**.



7. Ahora vaya al archivo **MainActivity.kt** y agregue el siguiente código, deberá quedar tal cual como se ve a continuación:

```
package net.lrivas.juegocirculos

import android.content.Intent
import android.os.Bundle
import android.widget.Button
import androidx.appcompat.app.AppCompatActivity

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

        findViewById<Button>(R.id.ButtonIniciar).setOnClickListener {
            startActivity(Intent(this, JuegoActivity::class.java))
        }
    }
}
```

8. Ahora crea una nueva actividad y colóquese el nombre de **JuegoActivity.kt**, y agrega el siguiente código.

```
package net.lrivas.juegocirculos
import android.content.Intent
import android.os.Bundle
import android.os.Handler
import android.os.Looper
import android.view.View
import android.widget.GridLayout
import android.widget.TextView
import androidx.appcompat.app.AppCompatActivity
import androidx.core.content.ContextCompat
```

```

class JuegoActivity : AppCompatActivity() {
    private lateinit var cuadrilla: GridLayout
    private lateinit var textoNivel: TextView
    private lateinit var circulos: List<View>
    private val manejador = Handler(Looper.getMainLooper())
    private var puntaje = 0
    private var nivelActual = 1
    private var circulosPorNivel = 5
    private var circulosVerdes = 0

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

        cuadrilla = findViewById(R.id.gridLayout)
        textoNivel = findViewById(R.id.levelText)
        circulos = List(12) { crearCirculo() }
        circulos.forEach { cuadrilla.addView(it) }

        iniciarNivel()
    }

    private fun crearCirculo(): View {
        return View(this).apply {
            layoutParams = GridLayout.LayoutParams().apply {
                width = 0
                height = 0
                columnSpec = GridLayout.spec(GridLayout.UNDEFINED, 1f)
                rowSpec = GridLayout.spec(GridLayout.UNDEFINED, 1f)
            }
            background = ContextCompat.getDrawable(this@JuegoActivity,
R.drawable.circle_red)
            setOnClickListener { alHacerClickEnCirculo(this) }
        }
    }

    private fun iniciarNivel() {
        textoNivel.text = "Nivel $nivelActual"
        puntaje = 0
        circulosVerdes = 0
        circulosPorNivel = 5 + (nivelActual - 1) * 2

        reiniciarCirculos()
        mostrarProximoCirculoVerde()
    }

    private fun reiniciarCirculos() {
        circulos.forEach { it.background =
ContextCompat.getDrawable(this, R.drawable.circle_red) }
    }

    private fun mostrarProximoCirculoVerde() {
        if (circulosVerdes < circulosPorNivel) {
            manejador.postDelayed({
                cambiarCirculoAleatorio()
                circulosVerdes++
                mostrarProximoCirculoVerde()
            }, 1000)
        } else {
            if (nivelActual == 2) {

```

```

        finalizarJuego(true)
    } else {
        nivelActual++
        iniciarNivel()
    }
}

private fun cambiarCirculoAleatorio() {
    val circulosRojos = circulos.filter {
        it.background.constantState == ContextCompat.getDrawable(this,
            R.drawable.circle_red)?.constantState
    }
    if (circulosRojos.isNotEmpty()) {
        val circuloAleatorio = circulosRojos.random()
        circuloAleatorio.background =
            ContextCompat.getDrawable(this, R.drawable.circle_green)
    }
}

private fun alHacerClickEnCirculo(vista: View) {
    if (vista.background.constantState ==
        ContextCompat.getDrawable(this,
            R.drawable.circle_green)?.constantState) {
        puntaje++
        vista.background = ContextCompat.getDrawable(this,
            R.drawable.circle_red)
    } else {
        finalizarJuego(false)
    }
}

private fun finalizarJuego(completado: Boolean) {
    manejador.removeCallbacksAndMessages(null)
    val intent = Intent(this, ResultadosActivity::class.java)
    intent.putExtra("PUNTAJE", puntaje)
    intent.putExtra("COMPLETADO", completado)
    intent.putExtra("NIVEL", nivelActual)
    startActivity(intent)
    finish()
}
}

```

9. Ahora crea el archivo activity_juego.xml y agregue el siguiente código.

```

<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
    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"
    android:background="#FFFFFF">

    <TextView
        android:id="@+id/headerText"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:background="@color/black"
        android:padding="16dp"
        android:text="Juego de Circulos"
    >

```



```

        android:textAlignment="center"
        android:textColor="#FFFFFF"
        android:textSize="24sp"
        android:textStyle="bold"
        app:layout_constraintTop_toTopOf="parent" />

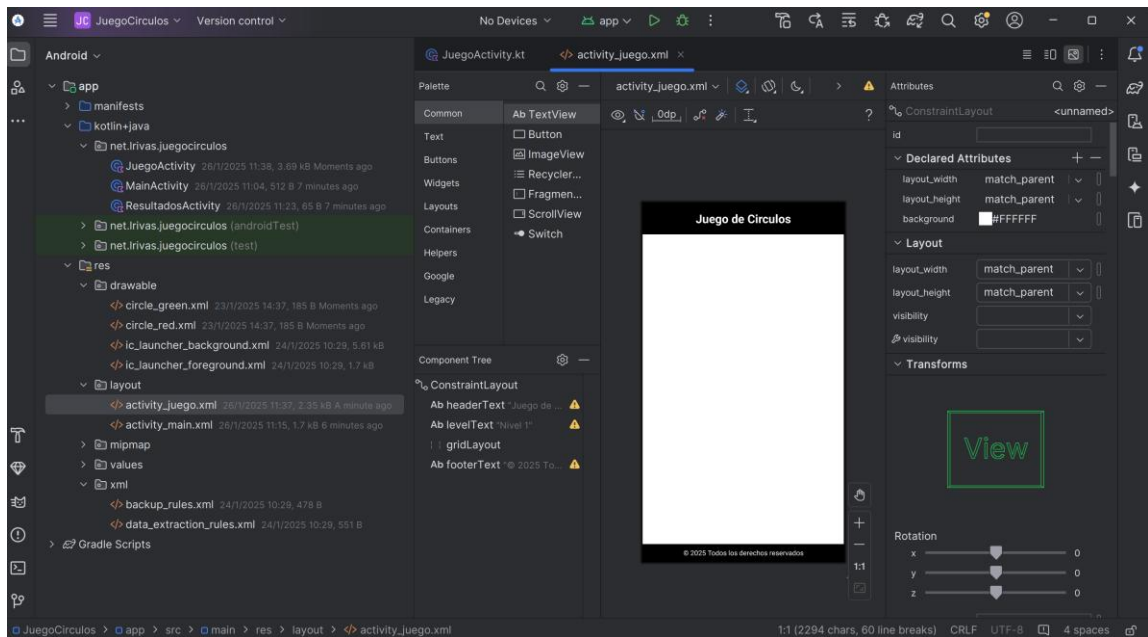
<TextView
    android:id="@+id/levelText"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="16dp"
    android:text="Nivel 1"
    android:textColor="@color/black"
    android:textSize="20sp"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@id/headerText" />

<GridLayout
    android:id="@+id/gridLayout"
    android:layout_width="345dp"
    android:layout_height="416dp"
    android:layout_margin="16dp"
    android:columnCount="3"
    android:padding="18dp"
    android:rowCount="4"
    android:useDefaultMargins="true"
    app:layout_constraintBottom_toTopOf="@id/footerText"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@id/levelText" />

<TextView
    android:id="@+id/footerText"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:background="@color/black"
    android:padding="8dp"
    android:text="© 2025 Todos los derechos reservados"
    android:textAlignment="center"
    android:textColor="#FFFFFF"
    android:textSize="14sp"
    app:layout_constraintBottom_toBottomOf="parent" />
</androidx.constraintlayout.widget.ConstraintLayout>

```

10. Le deberá mostrar un diseño como el siguiente.



11. Como siguiente paso vamos a crear el archivo ResultadosActivity.kt, en donde se mostrarán los resultados obtenidos en el juego.

```
package net.lirivas.juegocirculos

import android.content.Intent
import android.os.Bundle
import android.widget.Button
import android.widget.TextView
import androidx.appcompat.app.AppCompatActivity

class ResultadosActivity : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_resultados)

        val puntaje = intent.getIntExtra("PUNTAJE", 0)
        val completado = intent.getBooleanExtra("COMPLETADO", false)
        val nivel = intent.getIntExtra("NIVEL", 1)

        val textoResultado =
            findViewById<TextView>(R.id.resultadoText)
        val textoPuntaje = findViewById<TextView>(R.id.puntajeText)

        if (completado) {
            if (nivel > 2) {
                textoResultado.text = ";Felicitades! Completaste el juego"
            } else {
                textoResultado.text = ";Nivel superado!"
            }
        } else {
            textoResultado.text = ";Perdiste!"
        }
        textoPuntaje.text = "Puntaje: $puntaje"

        findViewById<Button>(R.id.nuevoJuego).setOnClickListener {
            startActivity(Intent(this, MainActivity::class.java))
        }
    }
}
```

```

        finish()
    }
}

```

12. Seguidamente se deberá crear el `activity_resultados.xml`, en el cual debes agregar el siguiente código.

```

13. <?xml version="1.0" encoding="utf-8"?>
    <androidx.constraintlayout.widget.ConstraintLayout
        xmlns:android="http://schemas.android.com/apk/res/android"
        xmlns:app="http://schemas.android.com/apk/res-auto"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:background="#FFFFFF">

        <TextView
            android:id="@+id/headerText"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:background="@color/black"
            android:padding="16dp"
            android:text="Juego de Circulos"
            android:textAlignment="center"
            android:textColor="#FFFFFF"
            android:textSize="24sp"
            android:textStyle="bold"
            app:layout_constraintTop_toTopOf="parent" />

        <TextView
            android:id="@+id/resultadoText"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginTop="32dp"
            android:text="¡Juego terminado!"
            android:textColor="@color/black"
            android:textSize="28sp"
            android:textStyle="bold"
            app:layout_constraintEnd_toEndOf="parent"
            app:layout_constraintStart_toStartOf="parent"
            app:layout_constraintTop_toBottomOf="@id/headerText" />

        <TextView
            android:id="@+id/puntajeText"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginTop="16dp"
            android:text="Puntaje: 0"
            android:textColor="@color/black"
            android:textSize="24sp"
            app:layout_constraintEnd_toEndOf="parent"
            app:layout_constraintStart_toStartOf="parent"
            app:layout_constraintTop_toBottomOf="@id/resultadoText"
        />

        <Button
            android:id="@+id/nuevoJuego"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"

```

```

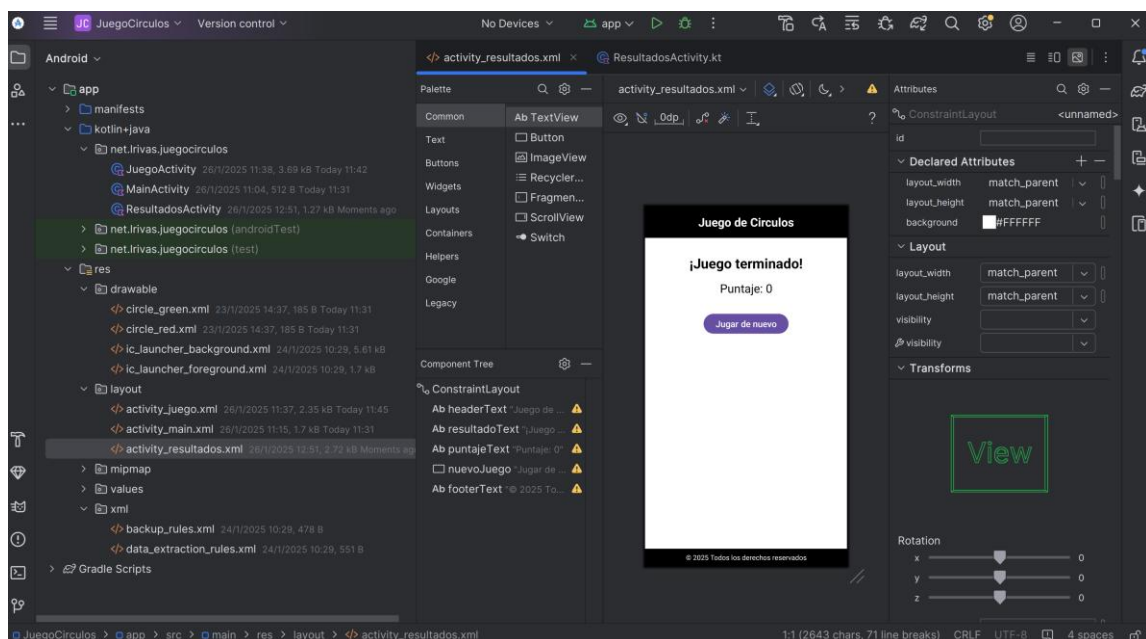
        android:layout_marginTop="32dp"
        android:text="Jugar de nuevo"
        android:textSize="18sp"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@id/puntajeText" />

<TextView
    android:id="@+id/footerText"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:background="@color/black"
    android:padding="8dp"
    android:text="© 2025 Todos los derechos reservados"
    android:textAlignment="center"
    android:textColor="#FFFFFF"
    android:textSize="14sp"
    app:layout_constraintBottom_toBottomOf="parent" />

</androidx.constraintlayout.widget.ConstraintLayout>

```

14. Te Deberá mostrar una imagen como la siguiente.



15. No olvides agregar tus nuevas activitys al AndroidManifest.xml, el cual te debvera quedara si:

```

<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools">

    <application
        android:allowBackup="true"
        android:dataExtractionRules="@xml/data_extraction_rules"
        android:fullBackupContent="@xml/backup_rules"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"

```

```

        android:supportRtl="true"
        android:theme="@style/Theme.JuegoCirculos"
        tools:targetApi="31">
        <activity
            android:name=".MainActivity"
            android:exported="true">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

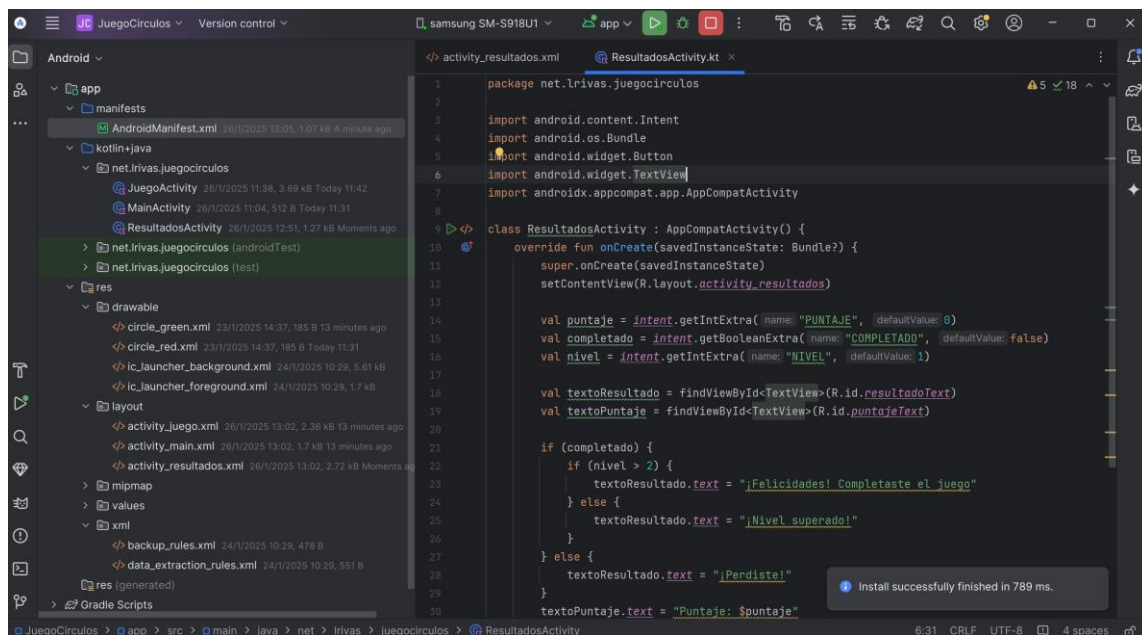
                <category
                    android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>

        <activity android:name=".JuegoActivity" />
        <activity android:name=".ResultadosActivity" />
    </application>

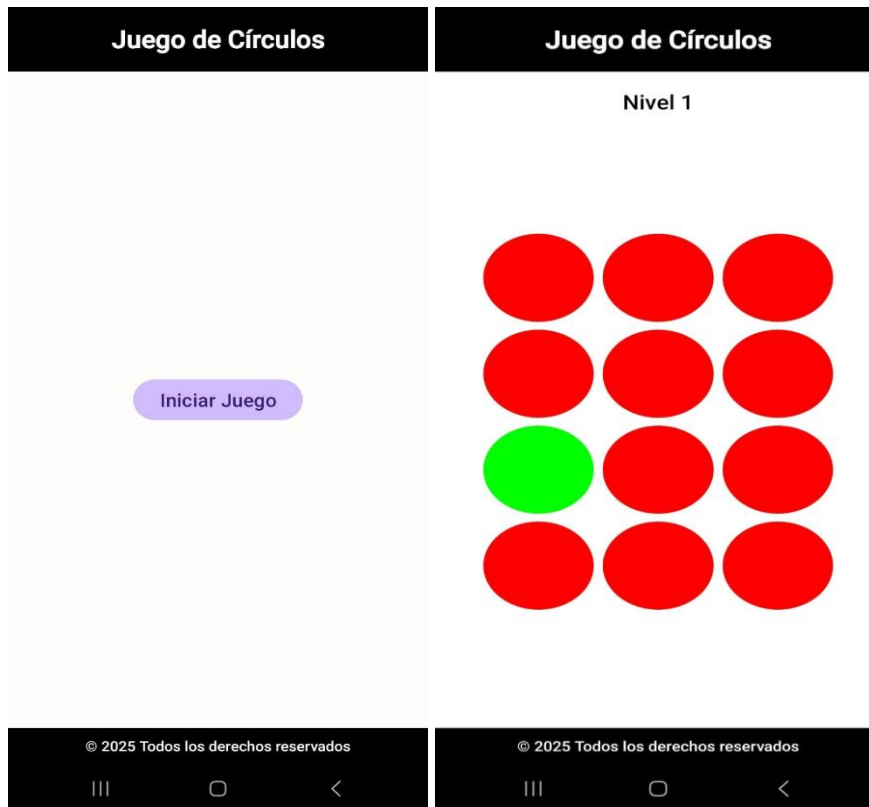
</manifest>

```

16. Una vez que este completado el código, realice el proceso de compilación e instalación de la aplicación:



17. Una vez el proceso haya terminado la aplicación deberá verse de la siguiente manera:



Listo ahora solo da clic en el botón Iniciar Juego y comienza el juego.

PRÁCTICA A REALIZAR

NOMBRE: Juego de Círculos

Indicación: Haciendo uso del ejemplo de la práctica y lo visto en clases, se requiere que dé respuesta a las preguntas además deberá desarrollar la siguiente aplicación:

1. Explique con sus propias palabras para que sirve el siguiente fragmento de código en la aplicación explicada en la práctica.

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

    cuadrilla = findViewById(R.id.gridLayout)
    textoNivel = findViewById(R.id.levelText)
    círculos = List(size: 12) { crearCirculo() }
    círculos.forEach { cuadrilla.addView(it) }

    iniciarNivel()
}
```

2. Explique brevemente que proceso realiza el siguiente código.

```
private fun mostrarProximoCirculoVerde() {
    if (círculosVerdes < círculosPorNivel) {
        manejador.postDelayed({
            cambiarCirculoAleatorio()
            círculosVerdes++
            mostrarProximoCirculoVerde()
        }, delayMillis: 1000)
    } else {
        if (nivelActual == 2) {
            finalizarJuego(completado: true)
        } else {
            nivelActual++
            iniciarNivel()
        }
    }
}
```

3. Explique con sus propias palabras para que nos sirve el código que se está utilizando en el archivo ResultadosActivity.kt.

Ejercicio Practico

Crear un juego que contenga varios niveles de dificultad (mínimo 2 niveles) en los cuales el usuario pueda ir avanzando según sus aciertos, para ello tome de base lo visto en esta práctica y continúe con el juego según su propia creatividad, el diseño y desarrollo del mismo queda a su criterio propio.

Resultados esperados:

1. Diseño de interfaz amigable y apegada a los requerimientos.
2. Interactividad fluida y dinámica:
 - El juego debe responder rápidamente a las acciones del usuario, mostrando retroalimentación inmediata (como cambios de color, sonidos o animaciones) para indicar aciertos o errores.
3. Progresión de niveles bien definida:
 - Aumentar la dificultad a medida que el usuario avanza.
4. Elabore un documento en Word, que contenga:
 - a. La portada (con todas sus generalidades)
 - b. Capturas de pantalla de como quedó su aplicación. (Incluya las pantallas principales).
 - c. Haga un repositorio en git, suba el proyecto y adjunte una captura del repositorio con el respectivo enlace.
 - d. Convierta el documento de WORD a PDF y adjúntelo al buzón de tareas correspondiente.

FORMA DE ENTREGA: Se deberá enviar al buzón de tarea llamado **Programación Orientada a Objetos**, de manera individual. **(No importa si lo han trabajado en parejas o equipos, siempre cada uno deberá adjuntar su evidencia en el buzón).**

TIPO DE ENTREGA: Enviar un documento en **PDF o DOCX** al final de la semana antes de las **23:59**.

RUBRICA DE EVALUACIÓN

Indicación: a continuación, se establecen los criterios de evaluación para la actividad de la semana.

#	CRITERIO	PTS.
1	Entrega el proyecto en la fecha establecida.	2
2	Dio respuesta a todas las preguntas.	1.5

3	Diseña la interfaz del juego de acuerdo con el requerimiento.	1.5
4	Creatividad y dinamismo.	1
5	Funciona la interfaz de puntaje final de acuerdo con el requerimiento.	2
6	Sube el proyecto a git y anexa el enlace en el documento.	1
7	El documento está ordenado y sin errores de ortografía.	1