



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO  
FACULTAD DE CONTADURÍA Y ADMINISTRACIÓN

DIVISIÓN SISTEMA UNIVERSIDAD ABIERTA Y EDUCACIÓN A DISTANCIA

**SUAYED**  
Sistema Universidad Abierta y  
Educación a Distancia

## **PROGRAMACION DE DISPOSITIV.MOVILES**

### **UNIDAD 4**

#### **ACTIVIDAD 1**

**NOMBRE DEL ALUMNO:**

PAULINA RODRIGUEZ SAMPEDRO

**NOMBRE DEL ASESOR:**

CRISTIAN CARDOSO ARELLANO

**SEXTO SEMESTRE**

GRUPO: 9696

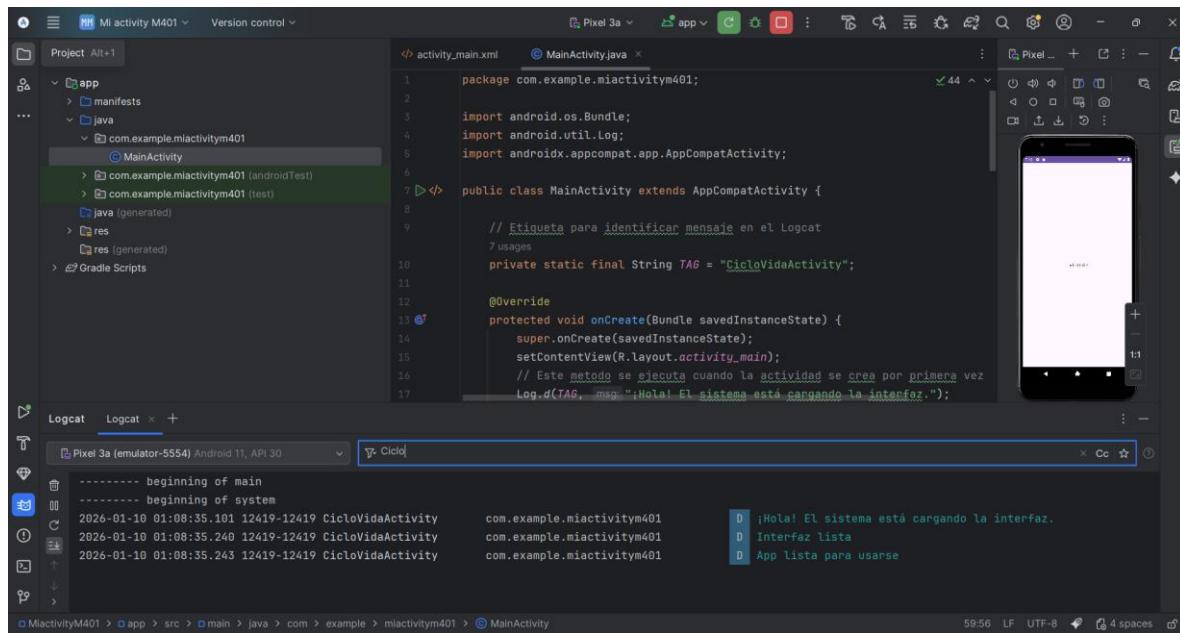
## CICLO DE VIDA DE UNA ACTIVITY

### Objetivo

Implementar una aplicación móvil funcional en Android Studio mediante el lenguaje Java, con el fin de comprender el ciclo de vida de una actividad a través del monitoreo de sus estados en la herramienta logcat, verificando cómo el sistema gestiona los recursos en tiempo real.

### Actividad

Ilustración 1. Apertura de la aplicación



En esta captura de pantalla se puede observar en el logcat que, al iniciar la actividad, el sistema pasa por tres estados, primero, onCreate inicializa los componentes y el diseño XML. Luego, onStart prepara la interfaz para ser visible, y finalmente onResume que permite que el usuario comience a interactuar con ella.

Ilustración 2. Inicio de la aplicación

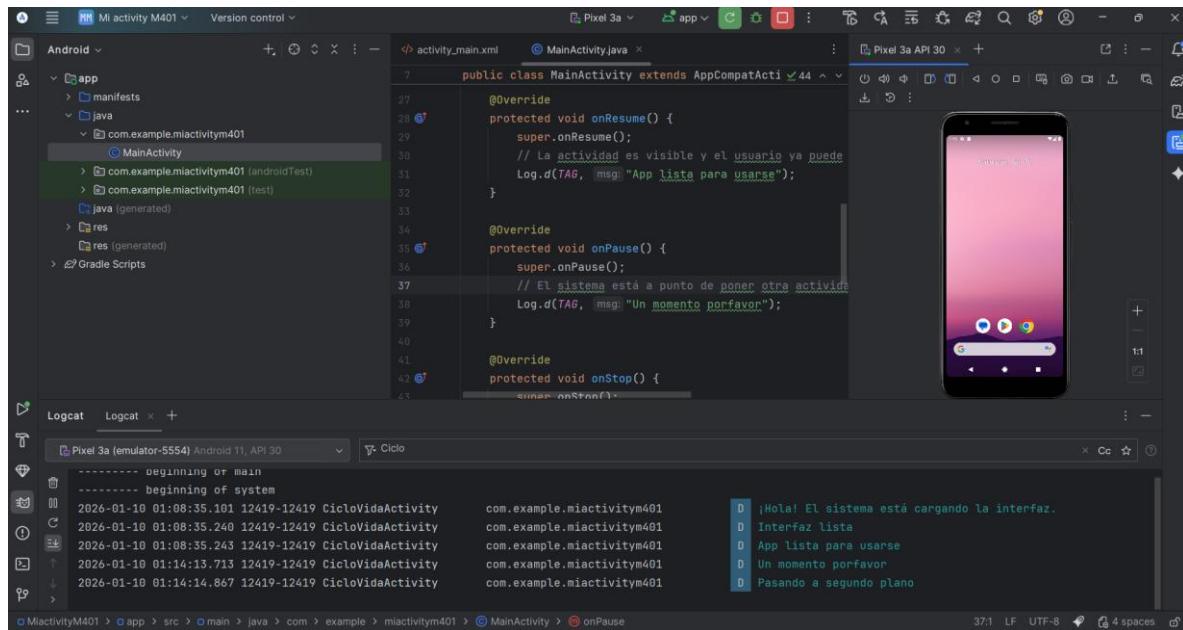
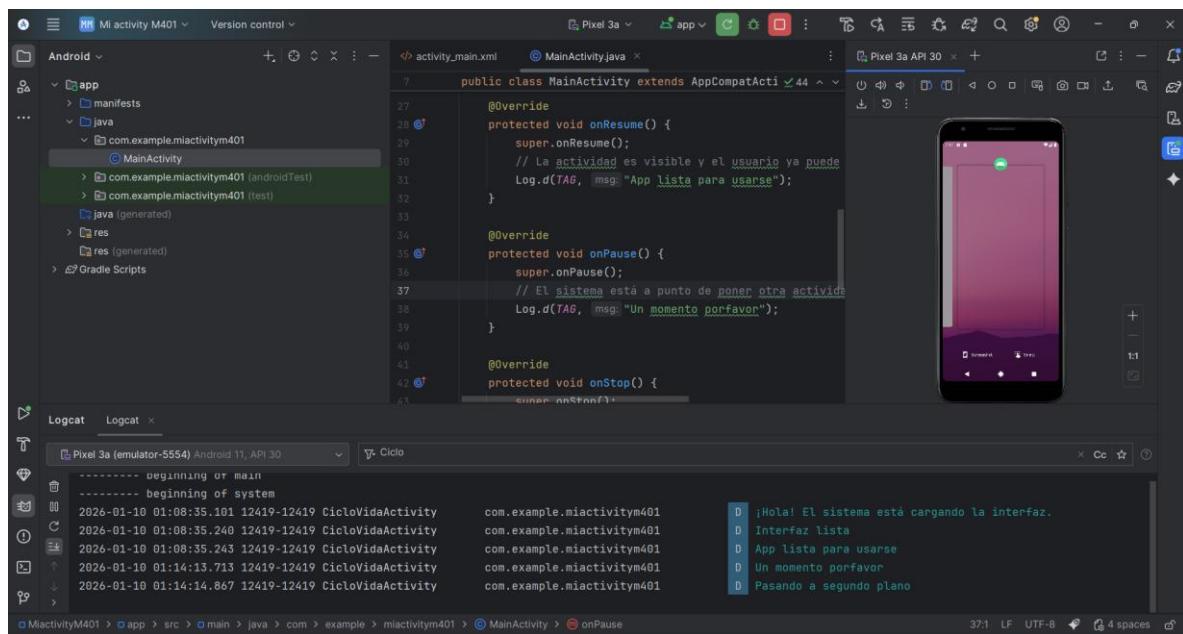
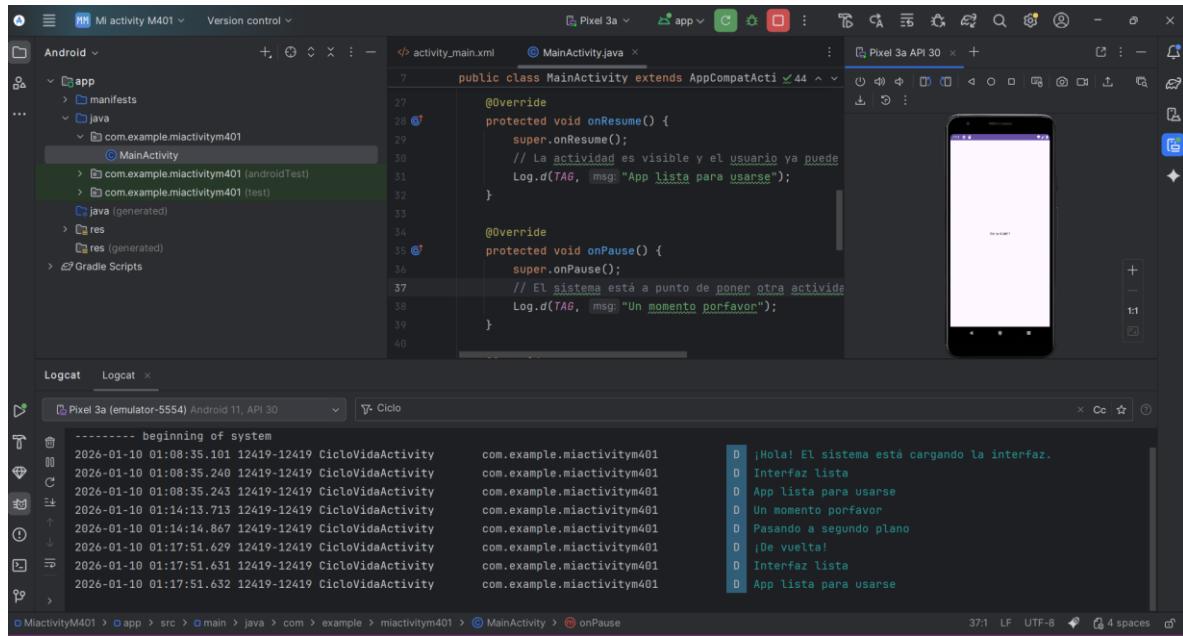


Ilustración 3. Apertura de un segundo plano



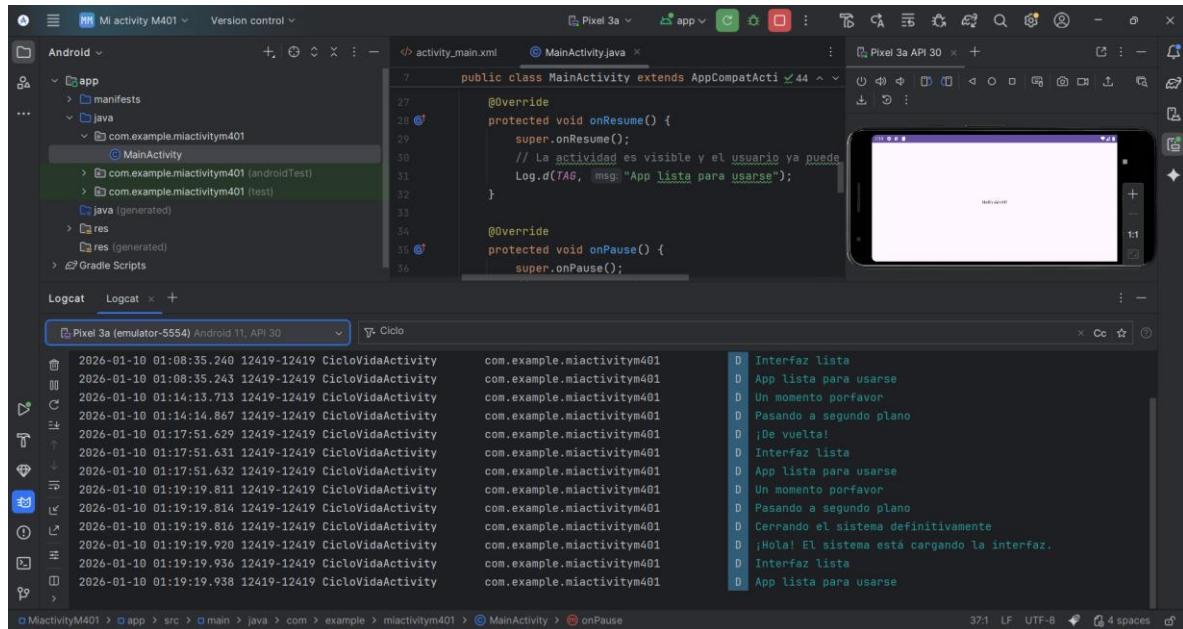
Al presionar el botón “home” se observa que la aplicación deja de estar en primer plano, el sistema ejecuta onPause para detener algunos procesos y onStop porque la actividad ya no es visible para el usuario. En este estado, la aplicación sigue viva en la memoria RAM, pero no consume ciclos de CPU innecesarios lo que optimiza la batería.

Ilustración 4. Retorno de la aplicación



Al abrir la aplicación de nuevo desde el menú de apps recientes, se observa que en el logcat aparece onRestart lo que indica que el sistema no necesita volver a crearla desde cero y en su lugar solo la reactiva. Además, el flujo salta directamente a onStart y onResume, recuperando el estado exacto donde el usuario la dejó.

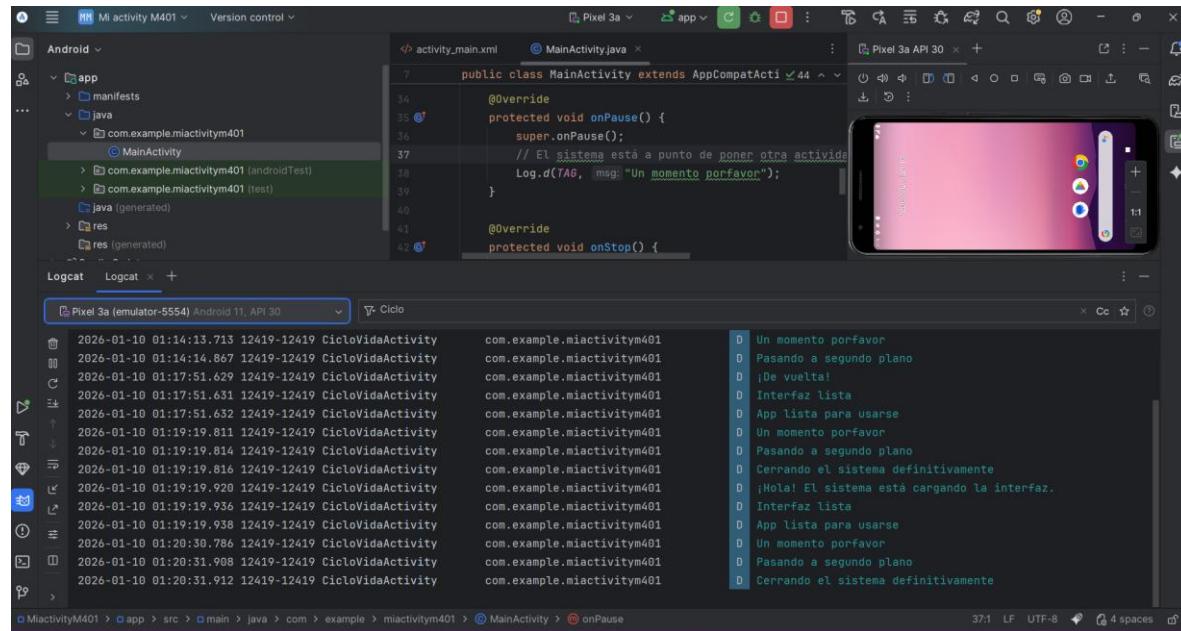
Ilustración 5. Rotación de pantalla



Al girar la pantalla, el sistema destruye la instancia actual de la actividad y crea una nueva para cargar los recursos correspondientes a la nueva orientación. En donde

el logcat muestra la secuencia completa de cierre (onDestroy) y reapertura inmediata (onCreate).

## Ilustración 6. Cierre de la aplicación



Una vez presionando el botón “atrás”, el logcat muestra onPause, onStop y onDestroy. Por lo que, al finalizar la sesión, la actividad recorre el camino de salida completo onDestroy siendo esta la última señal que recibe la actividad antes de ser eliminada por completo de la memoria RAM.

## Conclusión

Considero que la realización de esta actividad representa un pilar fundamental en mi formación académica, ya que permitió cerrar la brecha entre la teoría abstracta y la implementación técnica en un entorno real. Al aplicar los conceptos del ciclo de vida en Android Studio, se hace evidente que el desarrollo de software móvil no es un proceso lineal, sino un sistema reactivo que depende totalmente de la gestión de recursos del hardware. Comprender qué ocurre en el "detrás de escena" de un dispositivo le permite al profesional en informática anticipar errores críticos, evitar fugas de memoria y garantizar que la aplicación sea fluida, incluso cuando el sistema operativo debe priorizar otras tareas o ahorrar energía.

Dominar cada uno de los métodos del ciclo de vida de una actividad es crucial, comenzando por `onCreate`, donde se inicializan los componentes esenciales y el diseño; seguido de `onStart`, que prepara la interfaz para ser visible; y `onResume`, el estado crítico donde la app finalmente permite la interacción directa con el usuario. Asimismo, es vital entender métodos como `onPause`, que detiene procesos ligeros; `onStop`, que resguarda recursos cuando la actividad ya no es visible; `onRestart`, que permite recuperar el estado de la aplicación tras una pausa; y finalmente `onDestroy`,

que asegura la liberación definitiva de memoria antes de que la actividad sea eliminada.

### **Referencias**

Desarrolladores de Android. (s.f.). *Ciclo de vida de la actividad*.  
<https://developer.android.com/guide/components/activities/activity-lifecycle?hl=es-419#java>

Desarrolladores de Android. (s.f.). *Log*.  
<https://developer.android.com/reference/android/util/Log>