



**IPN**  
*Instituto Politécnico Nacional*



**UPIIZ**  
*Unidad Profesional Interdisciplinaria de Ingeniería  
Campus Zacatecas*

**Ingeniería en:**  
*Sistemas Computacionales*

**Materia:**  
*Desarrollo de Aplicaciones Móviles Nativas.*

**Docente:**  
*Ing. Efraín Arredondo Morales*

**Actividad:**  
*“Actividad II: Investigación o Infografía sobre el Ciclo de  
Vida de una Actividad”*

**Alumno:**  
*Alejandro Tamayo Castro*

**Grupo:**  
*7CM1*

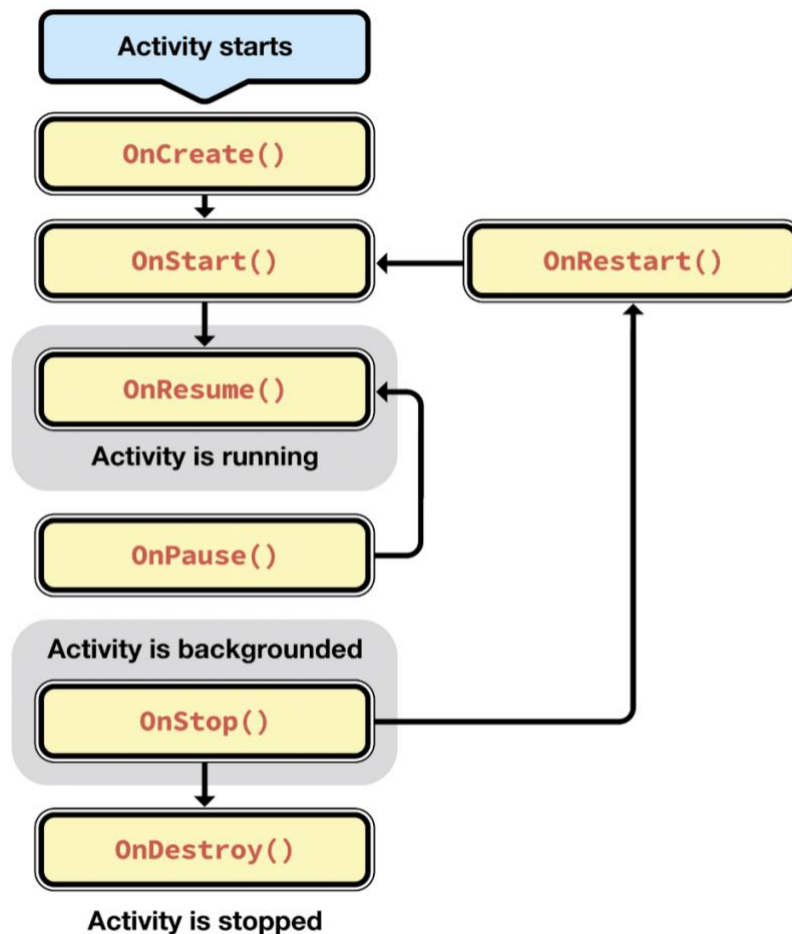
***Zacatecas, Zac. septiembre 19 de 2023***

Las actividades, de acuerdo con microsoft, son un concepto de programación exclusivo de Android, lo que son fundamentales en las aplicaciones de este sistema, las actividades pueden ser lanzadas por cualquier actividad de la aplicación. Siempre existe una actividad especificada como inicio de la aplicación, lo que permite que si ocurre un error y se finaliza la aplicación, esta pueda ser iniciada desde la ultima actividad registrada o abierta, además, el sistema puede pausar la aplicación si esta no se encuentra activa.

El ciclo de una actividad consta de varios métodos públicos de la clase *Activity* que proporcionan una administración de recursos al desarrollador. Esto permite cumplir con los *Estados de Actividad* que pueden ser:

- Running: Solo si están en primer plano, otorga prioridad alta y el sistema solo la cerrará en casos extremos.
- Paused: Si el dispositivo entra en suspension con una actividad visible, esta se oculta parcialmente, estas siguen activas y tienen la segunda prioridad más alta del sistema.
- Backgrounded: Actividades completamente ocultas por otra actividad, se consideran detenidas o en segundo plano, estas intentan conservar su información de estado durante el mayor tiempo posible, tienen el tercer nivel de prioridad y están debajo de las anteriores. El sistema eliminará primero estas actividades para obtener recursos para las actividades más prioritarias.
- Stopped: Actividades que son removidas de la memoria y por tanto, se tienen que reiniciar completamente cuando el usuario inicie las inicie, pueden restaurar su estado guardado anteriormente.

Los métodos del ciclo de vida de una actividad son 7, microsoft muestra el diagrama de flujo de la siguiente manera:



Al momento en que la actividad se inicia se comienza con el ciclo:

**OnCreate():** Este método es el primero que será llamado cuando se crea una actividad, en el se realiza la creación de vistas, inicialización de variables y el enlace de datos estáticos a listas. El método usa un parámetro Bundle para almacenar y pasar información de estado y objetos entre actividades. Si el lote no es null, la aplicación se está reiniciando y se restaurará su estado de la instancia anterior.

**OnStart():** Al finalizar OnCreate se llama este método, puede ser invalidado si es necesario realizar tareas específicas antes de que la actividad sea visible.

**OnResume():** Se llama cuando la actividad está lista para empezar a interactuar con el usuario, este método debe ser invalidado si es necesario aumentar fotografías, iniciar animaciones, escuchar actualizaciones de GPS, mostrar alertas o diálogos, conectar controladores de eventos, entre otras.

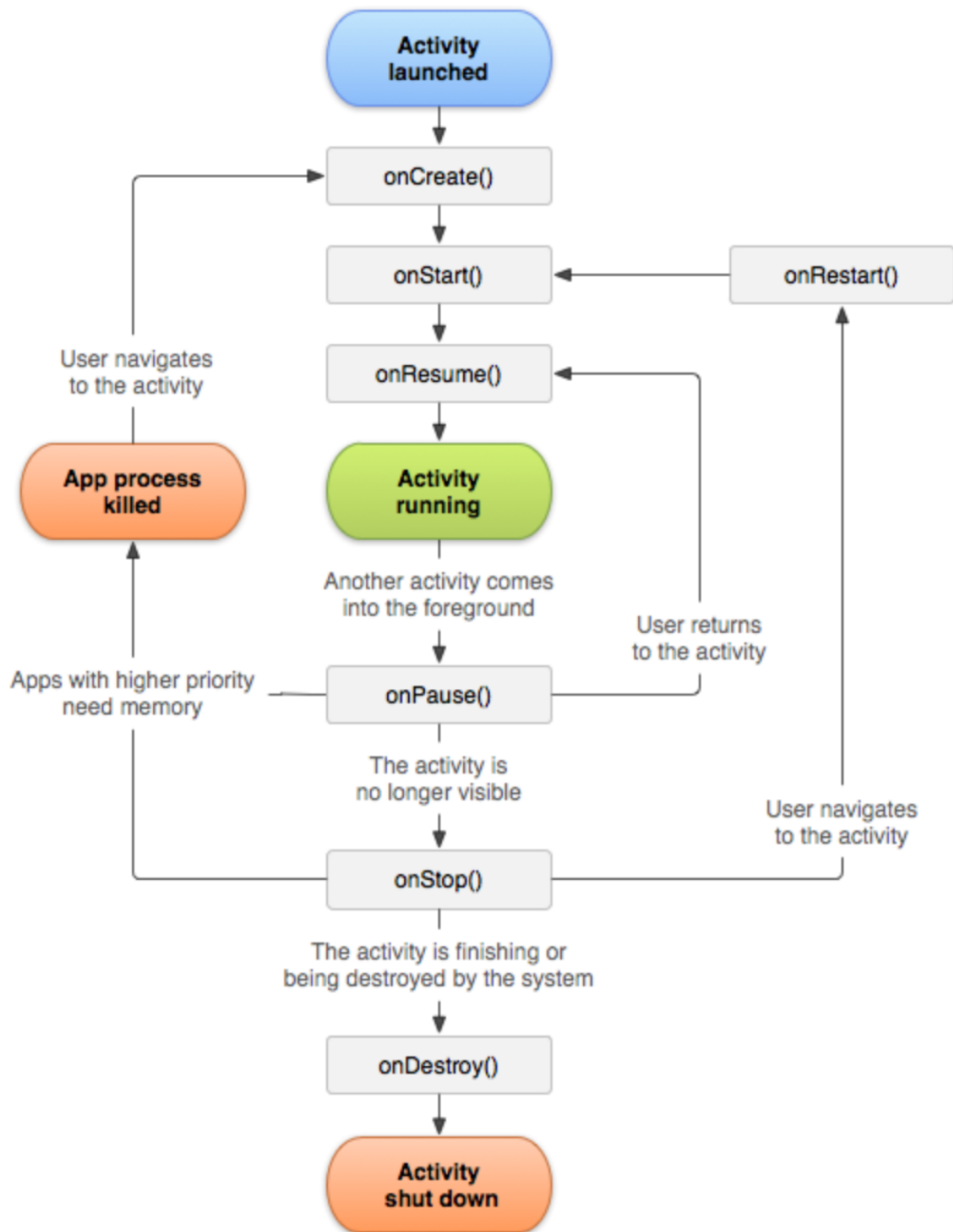
OnPause(): Este método es llamado cuando la actividad está a punto de ser colocada en segundo plano, este método debe ser invalidado si es necesario confirmar cambios no guardados, destruir o limpiar objetos, pausar animaciones, anular el registro de controladores de eventos o controladores de notificación o si la actividad está mostrando dialogos o alertas.

OnStop(): será llamado cuando la actividad ya no es visible para el usuario, si se esta iniciando una actividad nueva, se está llevando la actividad a primer plano o la actividad se está destruyendo.

OnDestroy(): Es el método final de una actividad antes de ser destruida y eliminada completamente de la memoria, este método puede no ser llamado si el sistema mata la actividad antes, este método puede ser invalidado para limpiar tareas de larga duración.

OnRestart(): este método es llamado despues de detener la actividad y antes de que se inicie nuevamente, este método no tiene reglas de uso o implementación pues onStart siempre será llamado cuando se crea o reiniciar la actividad.

Esta información no difiere con al presentada en el blog Developers de Android Studio, pues el diagrama de flujo que presentan es el siguiente:



Siendo que la información de cada método se encuentra más específica y con excepciones de versiones de Android, así como de su implementación con otros métodos para observar el ciclo de vida y/o administrar de mejor forma la memoria y batería del dispositivo.



<https://developer.android.com/guide/components/activities/activity-lifecycle?hl=es-419>

<https://learn.microsoft.com/es-es/xamarin/android/app-fundamentals/activity-lifecycle/>