

Multimedia y ciclo de vida de una aplicación (I)

Ciclo de Vida de una App

Ciclo de vida de una Aplicación (no Activity) en Android

Ciclo de vida de una Aplicación (I)

- ▶ A diferencia de otros SO **en Android, el usuario no decide cuando una aplicación termina.**
- ▶ Ésta **se mantiene en la pila de aplicaciones en ejecución, hasta que el sistema necesita memoria.**
- ▶ Una aplicación Android **corre dentro de su propio proceso Linux, se crea cuando necesita ser ejecutada.**
- ▶ **Continuará vivo el proceso hasta que el sistema necesite memoria para otra aplicación.**

Ciclo de vida de una Aplicación (II)

- ▶ Si tras eliminar el proceso de una aplicación, se vuelve a ella, **se crea de nuevo el proceso, perdiendo su estado.**
- ▶ **Ocurre lo mismo si giramos el teléfono.** Se crea un nuevo proceso con unas características gráficas diferentes.
- ▶ Android es sensible al ciclo de vida de una aplicación, por lo tanto, necesitamos comprender y manejar los eventos relacionados con el ciclo de vida si queremos crear aplicaciones estables.

Aplicaciones, Actividades y Procesos

- ▶ **Aplicación:** conjunto de *actividades y servicios* relacionados entre sí.
- ▶ **Actividad.** Cada una de las pantallas de la aplicación.
 - Desde una actividad podemos lanzar nuevas actividades de mi app o de otras apps.
 - El botón retroceder permite destruir la actividad actual y volver a la anterior.
- ▶ Cada aplicación se ejecuta en un **proceso linux distinto:**
 - De esta forma, **se delega en Linux tareas** tan vitales como **seguridad y multiproceso.**
 - **Todas las actividades y servicios de una app se ejecutan en el mismo proceso. UN PROCESO POR APLICACIÓN, NO POR ACTIVIDAD O SERVICIO.**

El ciclo de vida de una aplicación: Ejemplo paso a paso (I)

- Cuando encendemos un dispositivo, **inicialmente hay 2 procesos cargados en memoria:**

1. Proceso del Sistema

- Es el proceso más importante
- Se va a ir **encargando de ir lanzando nueva aplicaciones.**
- Lleva también la **cuenta de las actividades por las que va navegando el usuario.**
- Para ello, **usará una pila de Actividades**, que nos permitirá volver hacia actividades anteriores.
- En un primer momento, **ve que sólo tiene arrancada la actividad de Inicio** y por tanto, ha tenido que cargar una aplicación especial que es la **aplicación INICIO**. Por tanto, el 2º proceso es el de esta app.

2. Proceso de Inicio

- **Corresponde a la aplicación Inicio arrancada.**
- Desde ella podemos arrancar nuevas actividades. Por ejemplo, si pulsamos el botón correo, aparecerá **una nueva actividad.**



El ciclo de vida de una aplicación: Ejemplo paso a paso (II)

- Si, como dije antes, el **usuario pulsa en el icono CORREO** de la aplicación INICIO:

1. Le está diciendo al sistema que quiere ejecutar una nueva app: CORREO.
2. Más concretamente, que quiere arrancar la activity principal de esa aplicación, que es la BANDEJA DE ENTRADA.
3. Con lo cual:

- En la **pila de actividades**, aparece la actividad **BANDEJA DE ENTRADA** que muestra una lista de correos.
- **Aparece un nuevo proceso** en memoria correspondiente a la app **CORREO**.

4. Si el usuario pulsa sobre uno de esos correos para leerlo, hace que:

- Se genere una **nueva Actividad**, pero dentro de la misma aplicación CORREO.
- Veamos el gráfico correspondiente en la siguiente diapositiva



El ciclo de vida de una aplicación: Ejemplo paso a paso (III)

- **SITUACIÓN ACTUAL:** el usuario que estaba en la bandeja de entrada viendo su correo, ha pinchado sobre uno de ellos para leerlo y eso ha generado el siguiente gráfico:
 1. Se ha **creado una nueva actividad VISOR DE CORREO**, que aparecerá en la **pila de actividades**.
 2. **Como se trata de una actividad de la misma aplicación (la de correo electrónico), no es necesario crear un nuevo proceso.** Se ejecutan en el mismo proceso.
 3. Dentro de ese correo que está leyendo el usuario, puede haber un **enlace a una página web que le interese y puede decidir pinchar sobre él.**
 4. **Si el usuario pincha sobre ese enlace:**
 - El sistema va a crear una nueva actividad que pondrá en la pila de Actividades.
 - **Va a abrir el navegador web en un NUEVO PROCESO, dado que será una aplicación diferente.**
 - El gráfico correspondiente, lo tenemos en la siguiente diapositiva.



El ciclo de vida de una aplicación: Ejemplo paso a paso (IV)

- **SITUACIÓN ACTUAL:** el usuario que estaba leyendo un mensaje de correo ha visto un enlace que le interesaba a una página web y, ha pinchado sobre él, y eso ha generado el siguiente gráfico:



- **DE PRONTO:** se recibe una llamada de teléfono, esto hace que:
 1. El sistema lance una nueva aplicación (TELÉFONO) con su actividad correspondiente en la pila de actividades.
 2. Pero tenemos un problema: como se ve en el gráfico superior: **HEMOS AGOTADO LA MEMORIA** del dispositivo. No hay sitio para crear el nuevo proceso.

El ciclo de vida de una aplicación: Ejemplo paso a paso (V)

- **SITUACIÓN ACTUAL:** Hemos agotado la memoria y el sistema no puede crear el proceso TELÉFONO.

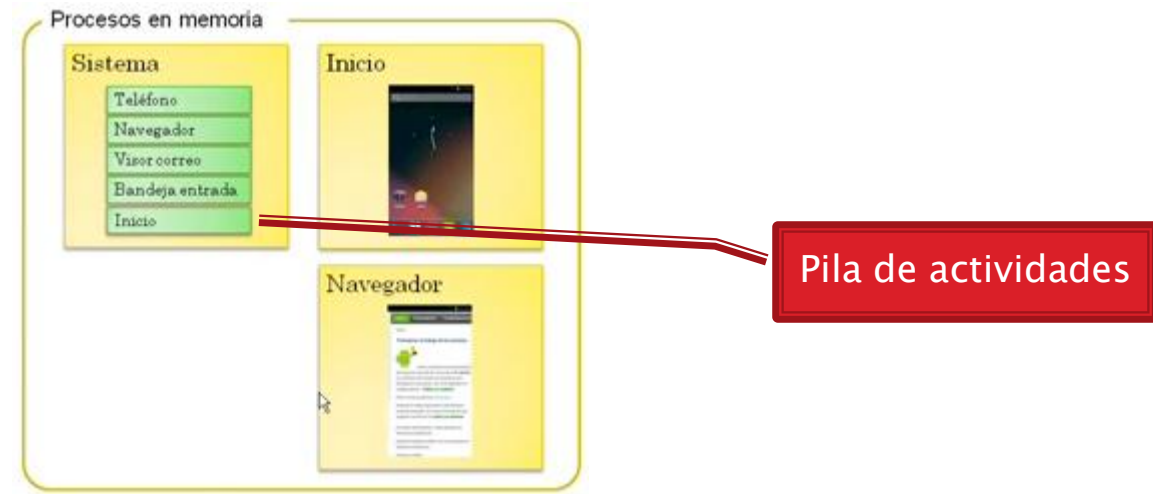
¿QUÉ TIENE QUE HACER EL SISTEMA?:

- **RESPUESTA:** Tiene que **entre todos los procesos** que tiene en este momento cargados, **eliminar uno de ellos**, intentando que sea la decisión más lógica.

¿CÓMO LO HACE?

- El navegador no sería buena idea porque es la app que estamos usando actualmente.
- Tampoco se puede eliminar el proceso del sistema.
- Tampoco se puede eliminar el proceso de INICIO, porque es el más importante para ir navegando por las diferentes aplicaciones del dispositivo.
- Por tanto, **la decisión más lógica**, sería **eliminar el proceso de la aplicación de correo (matar la app de CORREO), con todas las actividades que tuviese arrancadas en ese momento.**
- **Gráficamente nos quedaría como indica la siguiente diapositiva:**

El ciclo de vida de una aplicación: Ejemplo paso a paso (VI)



- **AHORA YA TIENE MEMORIA PARA CREAR EL PROCESO NECESARIO PARA LA APP Teléfono y arrancar la actividad correspondiente:**



El ciclo de vida de una aplicación: Ejemplo paso a paso (VII)

- **SITUACIÓN ACTUAL:** El sistema ha creado el proceso teléfono y arrancado su actividad, con lo cual, el usuario puede atender la llamada correspondiente.
- Cuando el usuario termine la llamada podrá **pulsar el botón VOLVER** para continuar con lo que estaba haciendo antes, que era leer una página Web.
- El sistema lo que hará al pulsar VOLVER, será:

1. Quitará de la pila de actividades la de arriba del todo (Teléfono)
2. Mata la actividad de la aplicación Teléfono, pero deja el proceso en MEMORIA de la app.

Gráficamente nos quedará:



El ciclo de vida de una aplicación: Ejemplo paso a paso (VIII)



- **SITUACIÓN ACTUAL:** El sistema ha quitado de la pila de Actividades la del teléfono y estaremos trabajando con el navegador, puesto que es la Actividad con la que estábamos antes del teléfono (se sabe porque ahora es la superior en la pila de actividades).
- Fijaros que **EL PROCESO TELÉFONO NO HA SIDO ELIMINADO. ¿Por qué no? Si ya no tiene ninguna actividad dentro?**
- **RESPUESTA:** Android lo va a dejar en memoria en previsión de que si el usuario recibe una nueva llamada o quiere volver a usar la aplicación teléfono, sea más rápido su funcionamiento porque ya está en memoria.
 - Es una especie de **CACHE**.

El ciclo de vida de una aplicación: Ejemplo paso a paso (IX)

- Si el usuario ahora pulsa la tecla de volver, porque quiere salir del navegador, puesto que ya terminó de leer:
 1. El sistema quita la actividad de la pila de actividades
 2. Mata la actividad, **pero deja el proceso NAVEGADOR en memoria:**



- Si miramos el gráfico superior, vemos que lo siguiente que debe hacer el sistema es cargar de nuevo la Actividad que queda ahora en la parte superior, es decir, **EL VISOR DE CORREO**.
- Pero **TENEMOS UN PROBLEMA:**
 - **El proceso correspondiente a la actividad VISOR CORREO el sistema se lo ha argado antes, cuando no tenía memoria.**

¿QUÉ TIENE QUE HACER AHORA EL SISTEMA ENTONCES?

- **RESPUESTA:** Volverlo a arrancar, pero tiene otro PROBLEMA: no tiene memoria para arrancarlo (*mirar gráfico de abajo situación actual*).
- **SOLUCIÓN:** Tendrá que volver a elegir qué proceso mata:
 - Como hay varios que no tienen ninguna actividad, sería lo más adecuado.
 - Además **el de teléfono es el más antiguo y por tanto, el más lógico para matarlo.**
- Por tanto, **matará el proceso Teléfono, creará uno para la aplicación correo y arrancará la actividad VISOR CORREO.**
- **NUEVO PROBLEMA:** como el proceso Correo había sido matado y vuelto a crear. **Cuando se crea de nuevo un proceso, todas las variables, lo que es el estado interno de esta actividad, se habrá perdido.** Y si el usuario estaba en una determinada línea leyendo, es muy posible que vuelva a aparecer desde el principio.



El ciclo de vida de una aplicación: Ejemplo paso a paso (XI)

- **SOLUCIÓN:** Para evitar el problema expuesto en la anterior diapositiva, el programador debe encargarse de guardar el estado de las actividades antes de que se destruyan y cuando se regeneren, volver a recuperar ese estado guardado.
- Gráficamente nos encontramos así:



El ciclo de vida de una aplicación: Ejemplo paso a paso XII)

- **CUANDO EL USUARIO TERMINE DE LEER EL CORREO:** Dará el botón de volver hacia atrás y eso provocará que el sistema:
 - Quite la actividad VISOR CORREO de la pila de actividades
 - Matará esa actividad del proceso CORREO, **pero no matará ese proceso CORREO.**
 - **Y, ADEMÁS,** se **recreará** a la actividad que quede ahora en la parte superior de la pila que, casualmente es una **actividad de la app CORREO** también. **Puede haber perdido también su estado si el programador no lo tenía bien programado.**
 - Gráficamente tendremos:



El ciclo de vida de una aplicación: Ejemplo paso a paso XIII)

- **CUANDO EL USUARIO TERMINE DE LEER TODOS LOS CORREOS:** Dará el botón de volver hacia atrás y eso provocará que el sistema:
 - Quite la actividad **BANDEJA DE ENTRADA** de la pila de actividades
 - Matará esa actividad del proceso **CORREO**, pero no matará ese proceso **CORREO**.
 - Y, **ADEMÁS**, se **recreará** a la actividad que quede ahora en la parte superior de la pila que, en este caso será la **actividad de INICIO**.
 - Así , dejará en memoria 2 procesos vacíos y 2 no vacíos.



**¿Cómo decide Androd qué
proceso matar cuando no
tiene memoria?**

¿Qué proceso se elimina? (I)

- ▶ Como hemos comentado, **Android mantiene en memoria todos los procesos que quepan aunque estos no se estén ejecutando.**
- ▶ Una vez que la **memoria está** llena y el usuario decide ejecutar una nueva aplicación, **el sistema ha de determinar qué proceso de los que están en ejecución ha de ser eliminado.**
- ▶ **Android ordena los procesos en una lista jerárquica, asignándole a cada uno una determinada "importancia".**
- ▶ Esta lista se confecciona **basándose en :**
 - **los componentes de la aplicación que están corriendo** (actividades y servicios) y
 - **el estado de estos componentes.**

¿Qué proceso se elimina? (II)

- ▶ Para establecer esta jerarquía de importancia se distinguen los siguientes **tipos de procesos**:
 - **Proceso de primer plano**: (*Foreground process*)
 - Hospeda una actividad en la superficie de la pantalla y con la cual el usuario está interactuando (su método **onResume()** ha sido llamado).
 - Debería haber solo uno o unos pocos procesos de este tipo.
 - **Sólo serán eliminados como último recurso**, si es que la memoria está tan baja que ni siquiera estos procesos pueden continuar corriendo.
 - **Proceso visible**: (*Visible process*)
 - Hospeda una actividad que está visible en la pantalla, pero no en el primer plano (su método **onPause()** ha sido llamado).
 - Considerado **importante**, no será eliminado a menos que sea necesario para mantener los procesos de primer plano.
 - **Proceso de servicio**: (*Service process*)
 - Hospeda un servicio que ha sido inicializado con el método **startService()**.
 - Aunque estos procesos no son directamente visibles al usuario, generalmente están haciendo tareas que para el usuario son importantes (tales como reproducir un archivo mp3 o mantener una conexión con un servidor de contenidos).
 - El **sistema siempre tratará de mantener esos procesos corriendo**, a menos que los niveles de memoria comiencen a comprometer el funcionamiento de los procesos de primer plano o visibles.

¿Qué proceso se elimina? (III)

- **Proceso de fondo:** (*Background process*)
 - Hospeda una actividad que no es actualmente visible al usuario (su método **onStop()** ha sido llamado).
 - Si estos procesos son eliminados no tendrán un directo impacto en la experiencia del usuario.
 - Como, hay muchos de estos procesos, el sistema asegura que el último proceso visto por el usuario sea el último en ser eliminado.
- **Proceso vacío:** (*Empty process*)
 - No hospeda a ningún componente de aplicación activo.
 - La única razón para mantener ese proceso es tener un "caché" que permita mejorar el tiempo de activación en la próxima vez que un componente de su aplicación sea ejecutado.

Cuestiones relacionadas con el ciclo de vida de una app

Cuestiones ciclo vida apps

- ▶ Puesto que una app incluye **actividades, servicios y fragments**:
 - Repasar el CICLO DE VIDA DE LAS **ACTIVIDADES** (*ver diapositivas Activity*).
 - En **servicios** aparecerá el ciclo de vida de los mismos
 - En **fragments** tendréis el ciclo de vida de los mismos.