

ESCUELA DE SISTEMAS Y TECNOLOGÍAS

Transparencias de ANALISTA DE SISTEMAS
Edición 2020 - Materia: Aplicaciones Android

TEMA: Intents - IPC

Consideraciones

- Estas transparencias **no** tienen el objetivo de suplir las clases.
- Por tanto, serán **complementadas** con ejemplos, códigos, profundizaciones y comentarios por parte del docente.
- El **orden** de dictado de estos temas está sujeto a la consideración del docente.

Referencias

- Intents y filtros de intents:
 - ❖ <https://developer.android.com/guide/components/intents-filters.html>
- Intents comunes:
 - ❖ <https://developer.android.com/guide/components/intents-common.html>
- Interacción con otras apps:
 - ❖ <https://developer.android.com/training/basics/intents/index.html>
- Enviando datos simples a otras apps:
 - ❖ <https://developer.android.com/training/sharing/send.html>
- Recibiendo datos simples de otras apps:
 - ❖ <https://developer.android.com/training/sharing/receive.html>
- Intents de Google Maps:
 - ❖ <https://developers.google.com/maps/documentation/android-api/intents>

Agenda

- Intents
- Intens Comunes
- IPC (Inter Process Communication)
- Recibir Intents Implícitos

Intents (1)

- Representación abstracta de una operación que esta a punto de ser ejecutada.
- Se utilizan para iniciar una acción desde otro componente (o aplicación).
- Usos:
 - Iniciar una actividad
 - Desde una actividad de la aplicación, llamar a **startActivity()** con un objeto Intent como parámetro que representa la nueva actividad a “mostrar”.
 - En caso que la actividad a llamar devuelva un resultado, llamar a **startActivityForResult()**. La respuesta también vendrá en forma de Intent.

Intents (2)

- Usos (cont.):
 - Iniciar un servicio
 - Se llama a **startService()** con un Intent que describe el servicio a iniciar y contiene la información necesaria para que este ejecute correctamente.
 - En el caso de servicios cliente/servidor (binded) llamar a **bindService()**.
 - Enviar un mensaje broadcast
 - ¿Qué es un broadcast? Un mensaje que todos (aplicaciones en este caso) reciben o pueden recibir.
 - Hay varios métodos para enviar broadcast hacia el sistema: **sendBroadcast()**, **sendOrderedBroadcast()** y **sendStickyBroadcast()**.

Intents (3)

➤ Explícito

- El Intent a iniciar se especifica por nombre (completo o "fully-qualified").
- Típicamente los Intents que utilizamos para iniciar componentes de nuestra aplicación son de este tipo (sabemos el nombre de clase).
- Ejemplos:
 - Iniciar una actividad a raíz de interacción del usuario.
 - Iniciar un servicio para bajar un archivo en segundo plano.
- El sistema siempre inicia inmediatamente el Intent.

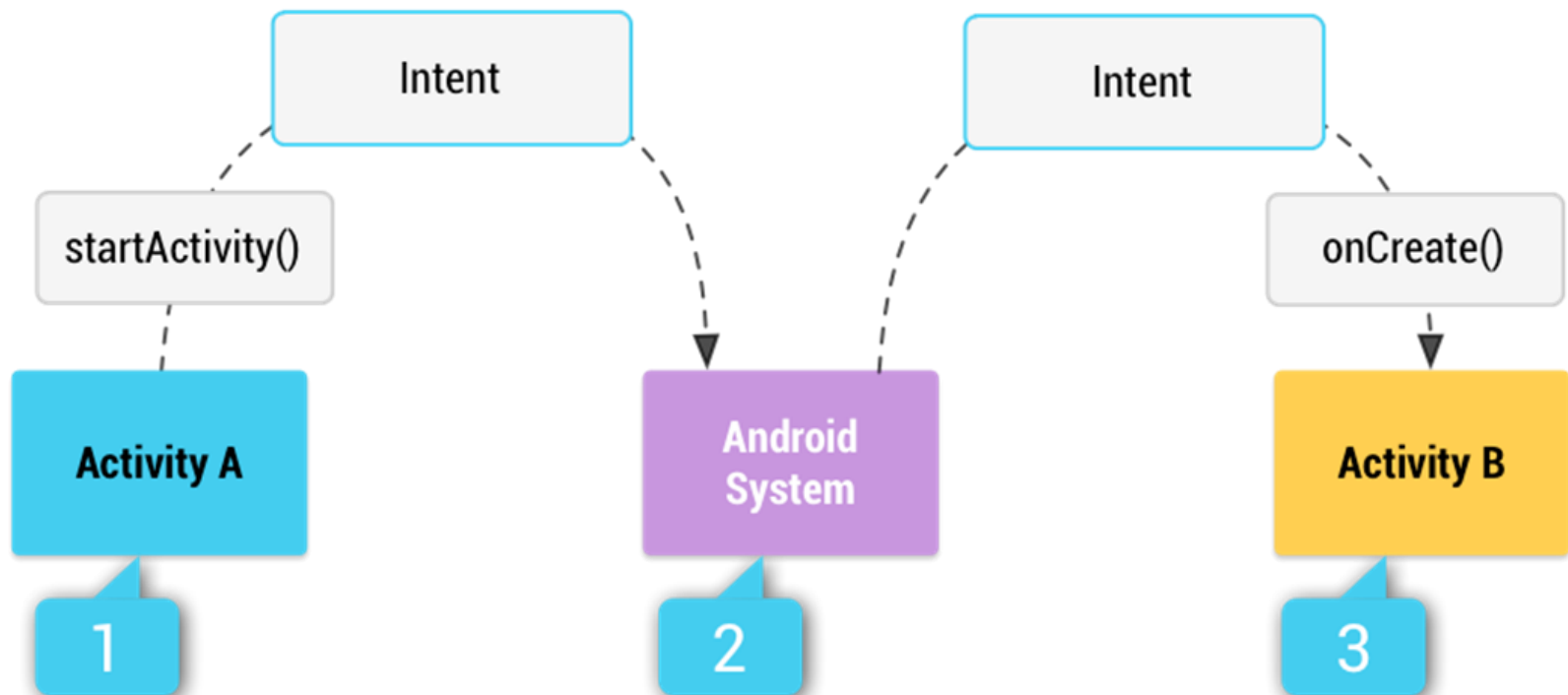
Intents (4)

➤ Implícito

- No se menciona explícitamente el componente a iniciar.
- En su lugar, se indica una acción a realizar.
- Los componentes que manejen esa acción, deberían responder al Intent (se muestra un selector si hay mas de uno). Para esto se usan los tags `<intent-filter>` del manifiesto de cada aplicación.
- Ejemplo:
 - Ver una URL, ver la ubicación en un mapa, etc.

Intents (5)

- Envío de un Intent IMPLÍCITO desde una Activity para crear otra Activity:



Intents Comunes (1)

- Intents implícitos de la plataforma Android:
 - Iniciar con **startActivity()** o **startActivityForResult()**.
- Se necesita proveer (dependiendo del Intent):
 - Action
 - Data URI
 - MIME Type
 - Extras
- Buena práctica
 - Utilizar **resolveActivity()** para Intents implícitos para poder controlar programáticamente si existe una aplicación que maneje la acción a realizar o no.

Intents Comunes (2)

- Intents desde Chrome: lanzados mediante URIs con cierto formato particular.

*<a
href="intent://scan/#Intent;scheme=zxing;package=com.google.
zxing.client.android;end"> Take a QR code *

Old: <iframe src="cutsomprotocol://page1"> </iframe>

Intents Comunes (3)

- Crear una alarma → ACTION_SET_ALARM
- Crear un cronometro → ACTION_SET_TIMER
- Mostrar todas las alarmas → ACTION_SHOW_ALARMS
- Agregar un evento al calendario → ACTION_INSERT
- Capturar y retornar una imagen → ACTION_IMAGE_CAPTURE
- Capturar y retornar un video → ACTION_VIDEO_CAPTURE
- Comenzar aplicación cámara en modo «still image» →
INTENT_ACTION_STILL_IMAGE_CAMERA
- Comenzar aplicación camara en modo video →
INTENT_ACTION_VIDEO_CAMERA
- Ver un contacto → ACTION_VIEW y URI=content:

Intents Comunes (4)

- Editar un contacto existente → ACTION_EDIT y URI=content:
- Agregar un contacto → ACTION_INSERT y URI=content:
- Crear un email con adjuntos opcionales
 - ACTION_SENDTO
 - ACTION_SEND (para un adjunto)
 - ACTION_SEND_MULTIPLE (para múltiples adjuntos)
- Recuperar un tipo específico de archivo
 - ACTION_GET_CONTENT
 - ACTION_OPEN_DOCUMENT
 - ACTION_CREATE_DOCUMENT
- Mostrar una ubicación en un mapa
 - ACTION_VIEW
- Cargar una URL → ACTION_VIEW y URI=http/s:<URL>

Intents Comunes (5)

- Reproducir archivo multimedia
 - ACTION_VIEW y URI=file: | content: | http: + <URI>
- Iniciar una llamada telefónica
 - ACTION_DIAL - abre aplicación telefono.
 - ACTION_CALL - (requiere permiso CALL_PHONE)
 - Data URI Scheme
 - tel:<phone-number>
 - voicemail:<phone-number>
- Crear un mensaje SMS/MMS con adjunto
 - ACTION_SENDTO
 - ACTION_SEND
 - ACTION_SEND_MULTIPLE

IPC (1)

- Inter Process Communication
- A veces es necesario proveer información a una Actividad para que sea ejecutada.
- También puede ser deseable esperar un resultado luego de la ejecución de una Actividad.
- Android posee mecanismos mas o menos complejos para la comunicación entre procesos.
 - Recordar 1: que 2 actividades que se están comunicando pueden pertenecer o no a la misma aplicación.
 - Recordar 2: que los procesos de aplicaciones Andoird están aislados uno del otro a nivel de Sistema Operativo (Linux Kernel).

IPC (2)

- Para proveer de información de “entrada” a una actividad, se carga la colección Extra del Intent asociado a la acción
- Para el caso en que se espera información de “salida” como producto de la ejecución de una actividad:
 - Se invoca a **startActivityForResult()** en lugar de **startActivity()** al momento de lanzar el Intent.
 - Se debe implementar el método **onActivityResult()** de la clase Activity para procesar la información requerida.
 - La actividad destino debe generar un Intent “hacia atrás” con resultado **RESULT_OK** y la información pedida en el mapa Extras pasándoselo al método **setResult()**.

Recibir Intents implícitos

- Para implementar una actividad que pueda recibir Intents implícitos:
 - Agregar nodos **<intent-filter>** en el archivo de manifiesto al nodo **<activity>** correspondiente.
 - Sub-nodos:
 - **<action>** - String literal nombre de la acción
 - **<data>** - tipo de datos aceptados
 - **<category>** - categoría del Intent
- Los Intents implícitos DEBEN ser definidos con la categoría **CATEGORY_DEFAULT**.