

ESCUELA DE SISTEMAS Y TECNOLOGÍAS

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

TEMA: User Interfaces II

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

- Documentación para developers:
 - ❖ [*http://developer.android.com/*](http://developer.android.com/)
- Notificaciones:
 - ❖ [*https://developer.android.com/guide/topics/ui/notifiers/notifications.html*](https://developer.android.com/guide/topics/ui/notifiers/notifications.html)
- Toasts:
 - ❖ [*https://developer.android.com/guide/topics/ui/notifiers/toasts.html*](https://developer.android.com/guide/topics/ui/notifiers/toasts.html)
- Fragmentos:
 - ❖ [*https://developer.android.com/guide/components/fragments.html*](https://developer.android.com/guide/components/fragments.html)
- Construyendo una IU dinámica con fragmentos:
 - ❖ [*https://developer.android.com/training/basics/fragments/index.html*](https://developer.android.com/training/basics/fragments/index.html)

Agenda

- Notificaciones
- Toasts
- Fragments

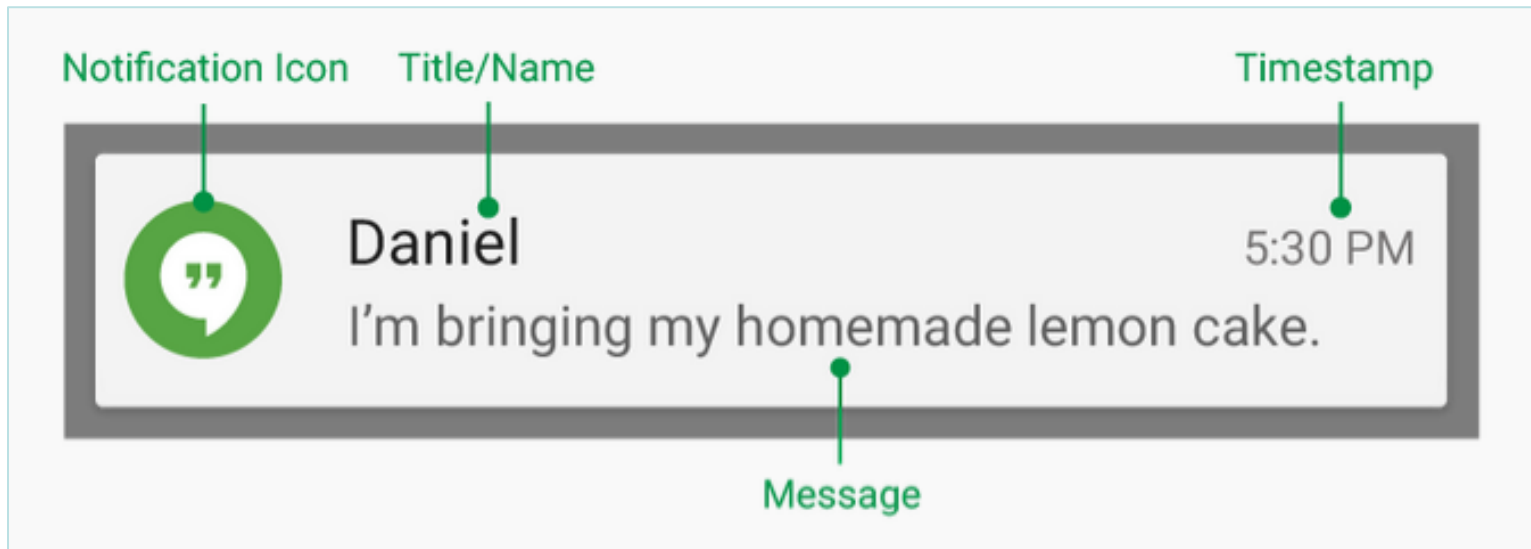
Notificaciones (1)

- Mensajes que las aplicaciones le envían al usuario.
- Se ejecutan fuera del contexto de la aplicación.
- Visibles en todo momento por el usuario
- Campos requeridos para su creación:
 - Icono pequeño (**setSmallIcon()**)
 - Titulo (**setContentTitle()**)
 - Texto detallado (**setContentText()**)
- Compatibilidad hacia atrás: **NotificationCompat.Builder** disponible en la biblioteca de compatibilidad de Android.
- Luego de creada, se la “envía” al sistema invocando al manejador **NotificationManager.notify(Notification)**.

Notificaciones (2)

➤ Layout

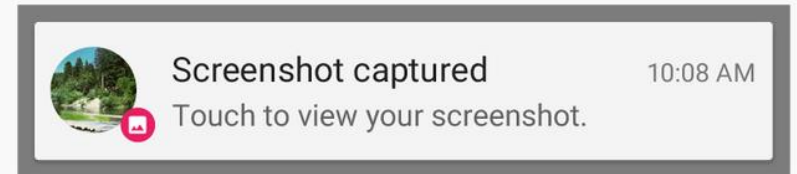
- Layout base conteniendo:
 - Icono
 - Titulo y Texto adicional
 - Marca de tiempo



Notificaciones (3)

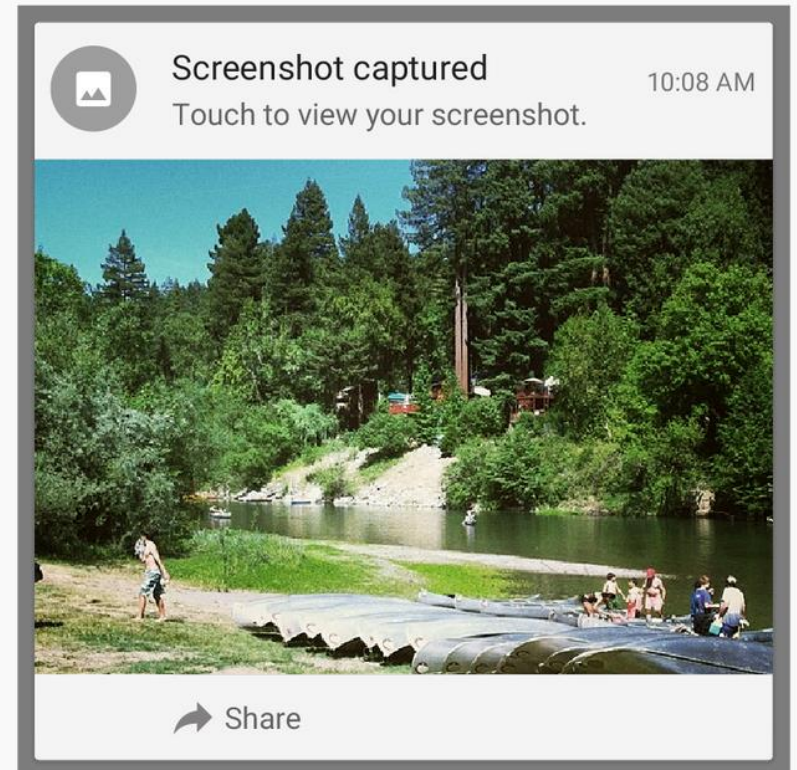
➤ Layout (cont.)

- Layout expandido:
 - Permite seleccionar cuánto detalle se muestra.
 - El usuario puede cambiar entre la vista expandida o compacta (para las notificaciones que lo soportan).



EXPAND ↓

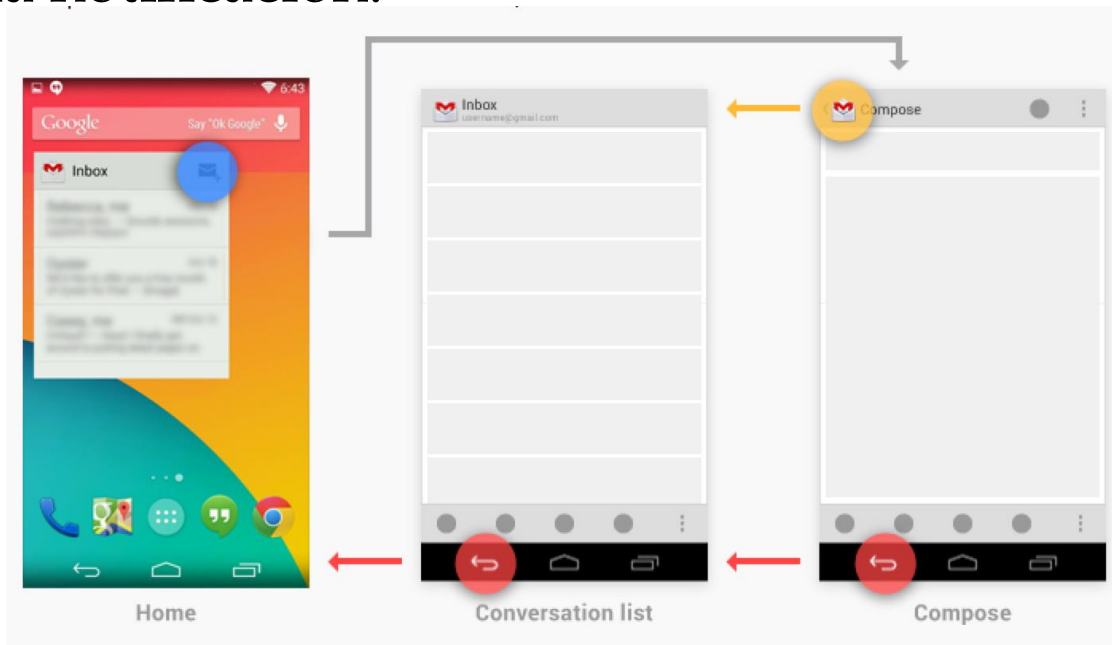
↑ CONTRACT



Notificaciones (4)

➤ Navegación

- Cuando el usuario toca dentro de la notificación, se aconseja llevarlo al punto más específico dentro la aplicación.
- También, que el usuario pueda tomar acciones en relación a la notificación.



Notificaciones (5)

➤ Acciones

- El usuario será llevado desde la notificación a una Actividad de la aplicación.
- Es posible agregar multiples acciones a una notificación.
 - Se debe especificar cuál es la acción por defecto (cuando el usuario simplemente toca la notificación).
 - Se puede agregar botones para las acciones (ver 'snooze' de las alarmas de Android).

Notificaciones (6)

➤ Android 5.0

- Notificaciones disponibles incluso en la pantalla de bloqueo (previo a desbloquear).
- Heads-up notification: de alta prioridad llegan incluso cuando el teléfono está en uso.
- Sincronización de notificaciones vía Cloud: Cuando una notificación se lee en un dispositivo se remueve de todos aquellos a los que esté logueado el usuario.

Notificaciones (7)

➤ Creación:

- **NotificationCompat.Builder()**
 - **setAction()**: para agregar acciones personalizadas
 - **setContentIntent()**: *PendingIntent* de la acción por defecto (evento de *touch*)
 - **setTitle()** / **setText()**
 - **setColor()** / **setSmallIcon()**
 - **setAutoCancel()**: indica si la notificación es automáticamente cancelada cuando se presiona
 - Luego se invoca método **build()** del Builder y nos devuelve la notificación a mostrar.
 - Finalmente, es necesario pasar la notificación al sistema para procesarla: **NotificationManager.notify()**

Notificaciones (8)

➤ Actualizar:

- Es posible actualizar notificaciones ya enviadas al sistema subyacente.
- Deben ser emitidas con un ID único al invocar a **NotificationManager.notify()**
- Si la notificación aún existe y sigue visible, el sistema actualiza el contenido.
- Si la notificación ha sido limpiada, una nueva es creada con el nuevo contenido.

Notificaciones (9)

➤ Remover:

- Es posible actualizar notificaciones ya enviadas al sistema subyacente.
- Las notificaciones permanecen visibles hasta que el usuario toma alguna acción.
- Si el usuario hace “Limpiar todas” en la barra de notificaciones, desea ignorar todas las notificaciones ya enviadas al sistema.
- Deberíamos hacer que nuestra aplicación las remueva para no molestar al usuario.

Notificaciones (10)

➤ Remover (cont.):

- **setAutoCancel()**: se limpia una vez que el usuario toca la notificación.
- **cancel()**: se invoca a cancelar para un ID específico de notificación.
- **cancelAll()**: remueve todas las notificaciones enviadas previamente por la aplicación al sistema.

Toast (1)

- Es una manera sencilla de proveer al usuario de información rápida.
- Pequeño popup que se acomoda al tamaño que ocupa el texto del mensaje.
- Desaparecen automáticamente luego de un tiempo predeterminado.
- En caso que se requiera una acción del usuario, siempre es mejor considerar el uso de Notificaciones.
- **Creación:**
 - Toast.makeText(Context, String, int)
 - Context: contexto de la aplicación
 - String: mensaje a desplegar
 - Int: duración. Toast.LENGTH_SHORT, Toast.LENGTH_LONG

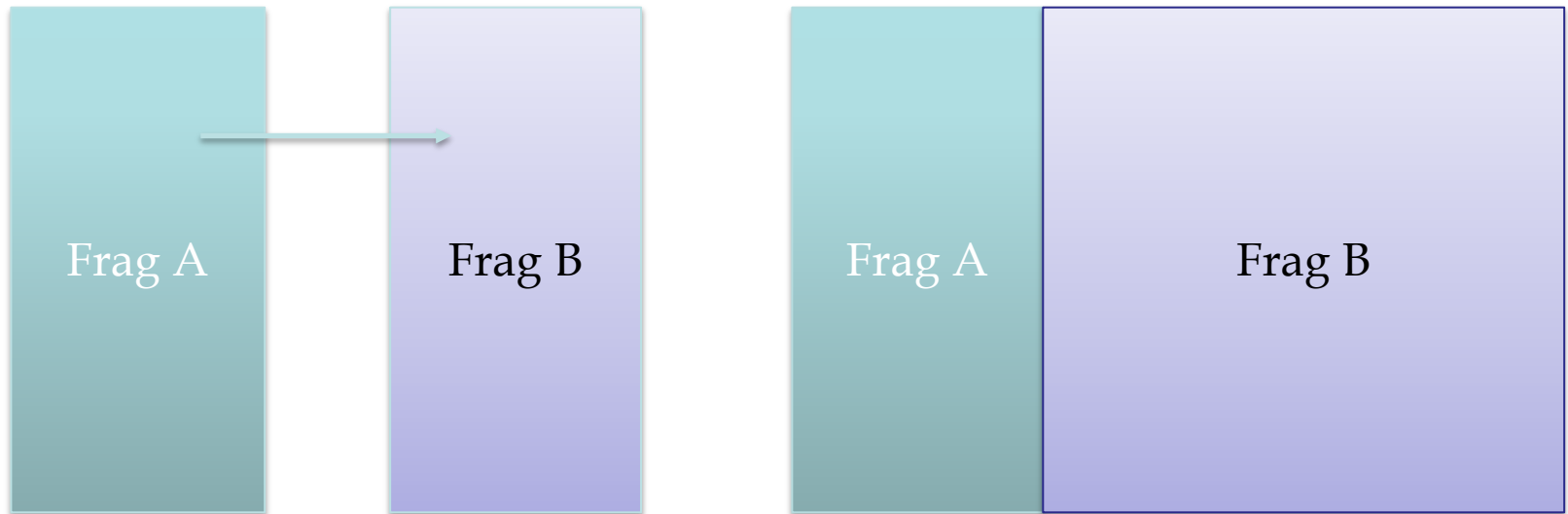
Toast (2)

➤ Personalizar:

- Alternativas de posicionamiento: a través del uso de la propiedad Gravity. Abajo, Arriba, Centrado, etc
- Vista especializada
 - Definir un Layout en XML
 - Crear (Inflater.inflate()) objeto View
 - Toast.setView(View)

Fragments (1)

- Los fragments son “mini-actividades” que se pueden reutilizar.
- El caso típico de uso es al desarrollar un “master - detail”.



Fragments (2)

➤ Funcionamiento:

- Al momento de usar fragments, definimos una vista que reemplazaremos por nuestro fragmento según necesitemos.
- Por ejemplo, si tenemos posibilidad de manejar dos columnas por estar en Tablet presentaremos la lista y detalle en la misma pantalla.
- Al comportarse de forma independiente, de forma similar a las actividades, tendremos un ciclo de vida.

Fragments (3)

