

Notificaciones (1)

Las notificaciones permiten a las aplicaciones informar al usuario sobre los eventos importantes: recepción de un mensaje, llamada perdida, evento de calendario, etc.

La barra de notificaciones se muestra en la parte superior de la pantalla, y podemos desplegarla para ver su contenido.



94

Notificaciones (2)

Debemos ser cuidadosos a la hora de elegir los eventos de los que vamos a informar, ya que un uso excesivo de notificaciones puede molestar al usuario y hacer que éste bloquee las notificaciones de la aplicación o incluso desinstale la aplicación.

Casos en que se recomienda utilizar notificaciones:

- Eventos asociados al tiempo y en interacción con otras personas: mensaje, email, agenda, etc.
- Eventos que informen al usuario de una alerta suscrita por la aplicación: noticias, resultados deportivos, etc.
- Ejecución de servicios con los que se puede interactuar: música, medidor de actividad, etc.

95

Notificaciones (3)

Casos en que no se deben utilizar notificaciones:

- Mensajes publicitarios.
- Disponibilidad de nuevas versiones.
- Información que no concierne de forma directa al usuario.
- Errores de aplicación.
- Ejecución de servicios con los que el usuario no puede interactuar.

96

Notificaciones (4)

■ Apariencia

Desde la versión 5 de Android, las notificaciones se componen de tres elementos:

- Icono de notificación: icono que representa a la aplicación o a algún evento determinado.
- Título de la notificación y descripción.
- *Timestamp*: momento de la recepción.



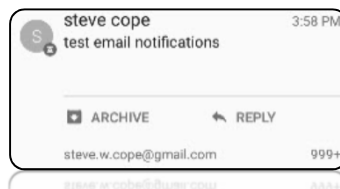
97

Notificaciones (5)

■ Apariencia

Las notificaciones se pueden extender para ver todo su contenido (por ejemplo, para ver el contenido íntegro de un mensaje) o agruparse (por ejemplo, cuando se reciben dos mensajes del mismo usuario)

Además, las notificaciones pueden incluir botones de acción para facilitar la interacción del usuario. Por ejemplo, al recibir un correo electrónico disponemos de botones para responder y archivar.



98

Notificaciones (6)

■ Implementación

Para que la notificación responda con una acción al pulsar sobre ella, hemos de definir un objeto *PendingIntent* mediante el método *getActivity* cuyos parámetros son:

- Contexto actual.
- Código identificador de la petición (un valor entero).
- *Intent* que representa la actividad que se iniciará.
- *Flag* que determina el modo de funcionamiento.

```
Intent nIntent = new Intent(MainActivity.this, MainActivity.class);
PendingIntent nPending = PendingIntent.getActivity
    (MainActivity.this, REQUEST_CODE, nIntent, PendingIntent.FLAG_ONE_SHOT);
```

En este ejemplo, al pulsar sobre la aplicación volveremos a la aplicación que generó la notificación.

99

Notificaciones (7)

■ Implementación

Los posibles valores del modo de funcionamiento son los siguientes:

- ***FLAG_CANCEL_CURRENT***: si el *intent* ya existe, será destruido al crear el nuevo.
- ***FLAG_NO_CREATE***: si el *intent* no existe, no se creará y el método devolverá *null*.
- ***FLAG_ONE_SHOT***: el *intent* sólo podrá utilizarse una vez.
- ***FLAG_UPDATE_CURRENT***: si el *intent* ya existe, será conservado y se actualizará.

100

Notificaciones (8)

■ Implementación

Una vez definido el *PendingIntent*, crearemos el constructor de notificación *NotificationCompat.Builder* y añadimos sus parámetros:

- ***setSmallIcon***: establece el icono (*drawable*).
- ***setContextTitle***: establece el título.
- ***setContextText***: establece la descripción.
- ***setWhen***: establece el *timestamp* (momento en que se recibe)
- ***setContentIntent***: establece la acción (*PendingIntent*).

```
NotificationCompat.Builder nBuilder =  
    new NotificationCompat.Builder(MainActivity.this);  
nBuilder.setWhen(System.currentTimeMillis());  
nBuilder.setSmallIcon(R.drawable.android_contacts);  
nBuilder.setContentTitle("Titulo notificación");  
nBuilder.setContentText("Texto de la notificación");  
nBuilder.setContentIntent(nPending);
```

Notificaciones (9)

■ Implementación

A continuación obtenemos la notificación utilizando el método *build()* sobre el constructor de notificación.

Y por último, enviamos la notificación. Para ello obtenemos una instancia del gestor de notificaciones (*NotificationManager*) mediante *getSystemService* (*NOTIFICATION_SERVICE*), y llamamos al método *notify* estableciendo como parámetros un id (entero) y la notificación.

```
int nId = 1;
Notification noti = nBuilder.build();
NotificationManager nManager =
    (NotificationManager) getSystemService(NOTIFICATION_SERVICE);
nManager.notify(nId, noti);
```

102

Notificaciones (10)

■ Implementación

Adicionalmente podemos añadir acciones a nuestra notificación, para ello hemos de utilizar el método *addAction* sobre el constructor de la notificación antes de llamar al método *build* para obtener la notificación.

El método *addAction* recibe como parámetros tres elementos: icono (*drawable*), título (*String*) y acción (*PendingIntent*).

```
nBuilder.addAction(R.drawable.contactos, "Contactos", pendingContactos);
```

103

Notificaciones (11)

■ Implementación (API 26 o superior)

A partir del nivel 26 de API (Android 8) al constructor de notificación (*NotificationCompat.Builder*) además del contexto hay que pasarle el identificador de un canal, que previamente hemos debido crear (*NotificationChannel*).

Para que la aplicación funcione independientemente del nivel de API, tendremos que realizar una comprobación del nivel de API y utilizar el tipo de constructor de notificación adecuado.

104

Notificaciones (12)

■ Implementación (API 26 o superior)

```
NotificationManager nManager =
    (NotificationManager) getSystemService(NOTIFICATION_SERVICE);
NotificationCompat.Builder nBuilder;
if (Build.VERSION.SDK_INT < 26) {
    nBuilder = new NotificationCompat.Builder(MainActivity.this);
} else {
    NotificationChannel nCanal = new NotificationChannel
        ("micanal", "nombreCanal", NotificationManager.IMPORTANCE_DEFAULT);
    nManager.createNotificationChannel(nCanal);
    nBuilder = new NotificationCompat.Builder(MainActivity.this, "micanal");
}
nBuilder.setWhen(System.currentTimeMillis());
nBuilder.setContentTitle("Nueva notificación");
nBuilder.setContentText("Texto completo de la notificación");
nBuilder.setSmallIcon(android.R.drawable.star_big_on);
nBuilder.setContentIntent(nPendingIntent);
nBuilder.addAction(R.drawable.phone, "Call", pendingCall);
Notification noti = nBuilder.build();
nManager.notify(nId, noti);
```

105