

Temari

- 1) Introducció
- 2) Framework Android
- 3) Projectes Android i Android SDK
- 4) Activity
- 5) Fragments, Views i ListViews
- 6) Intents
- 7) Layouts i Custom Views
- 8) Resources i Themes
- 9) Dialogs, Menus i WebView
- 10) Persistència de dades
- 11) Tasques en Background i internet
- 12) SQLite i content providers
- 13) Notificacions



13 – Notificacions

- Notificacions
- Crear una notificació bàsica
- Actualitzar i eliminar una notificació existent
- Pending intents
- Més opcions per les notificacions

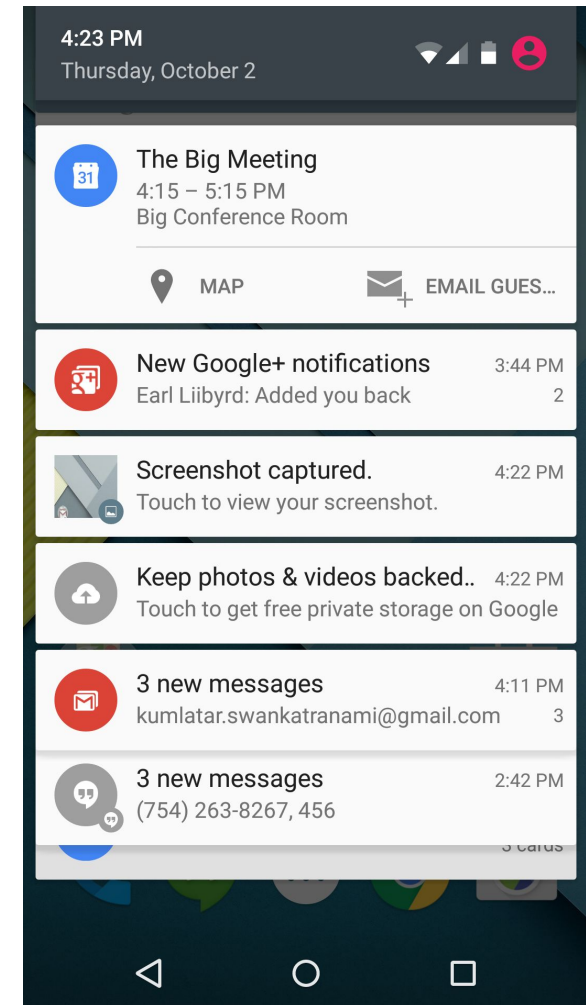
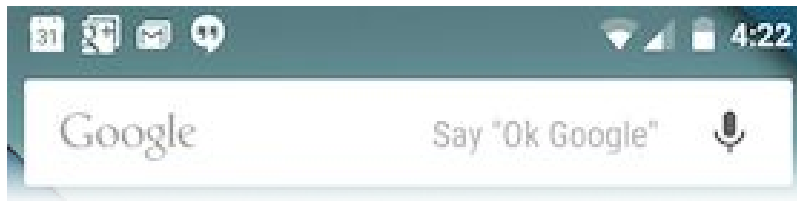


13 – Notificacions

Notificacions

Son missatges que es mostren fora de la UI de l'aplicacio.

Apareix primer una icona a la area de notificacions, que despres es pot expandir en obrir el **notification drawer**



13 – Notificaciones

Crear una notificació bàsica

S'utilitza la classe **NotificationCompat.Builder** per configurar una notificacio.

Ha de tenir **obligatoriament**:

- Icona amb **setSmallIcon()**
- Títol amb **setContentTitle()**
- Text amb **setContentText()**

```
NotificationCompat.Builder mBuilder =  
    new NotificationCompat.Builder(this)  
        .setSmallIcon(R.drawable.notification_icon)  
        .setContentTitle("My notification")  
        .setContentText("Hello World!");
```



13 – Notificaciones

Crear una notificació bàsica

Tot i que no es estrictament obligatori, es molt recomanable establir una **notification action** (que fer en clicar sobre la notificacio)

→ **PendingIntent** → **Intent** → **Activity**

```
Intent resultIntent = new Intent(this, ResultActivity.class);
...
// Because clicking the notification opens a new ("special") activity, there's
// no need to create an artificial back stack.
PendingIntent resultPendingIntent =
    PendingIntent.getActivity(
        this,
        0,
        resultIntent,
        PendingIntent.FLAG_UPDATE_CURRENT
    );
```



13 – Notificaciones

Crear una notificació bàsica

Despres es crida al metode **NotificationBuilder.build()** per crear un objecte **Notification**

Finalment es passa la notificacio al **NotificationManager**

```
NotificationManager mNotificationManager =  
    (NotificationManager) getSystemService(Context.NOTIFICATION_SERVICE);  
// mId allows you to update the notification later on.  
mNotificationManager.notify(mId, mBuilder.build());
```



13 – Notificaciones

Crear una notificació bàsica

Despres es crida al metode **NotificationBuilder.build()** per crear un objecte **Notification**

Finalment es passa la notificacio al **NotificationManager**

```
NotificationManager mNotificationManager =  
    (NotificationManager) getSystemService(Context.NOTIFICATION_SERVICE);  
// mId allows you to update the notification later on.  
mNotificationManager.notify(mId, mBuilder.build());
```



13 – Notificaciones

Actualizar o eliminar una notificació

En generar una notificacio, es passa un id de la notificacio.

Si l'id coincideix, no es crea una nova notificacio, si no que s'actualitza

```
// Sets an ID for the notification
int mNotificationId = 001;
// Gets an instance of the NotificationManager service
NotificationManager mNotifyMgr =
    (NotificationManager) getSystemService(NOTIFICATION_SERVICE);
// Builds the notification and issues it.
mNotifyMgr.notify(mNotificationId, mBuilder.build());
```

Amb aquest id es poden fer modificacions o cancel·lar la notificacio

- SetAutoCancel() per quan l'usuari la clica
- Cancel() per eliminar una notificacio (amb el id)
- CancelAll() per eliminar totes les que ha enviat la app



13 – Notificaciones

PendingIntent

Serveix per encapsular un Intent per a la seva execució posterior fora del Context on s'ha creat

→ permissions com la app que l'ha creat, encara que l'executi una altra

```
// In the MainActivity.onCreate()

int seconds = 3;
// Create an intent that will be wrapped in PendingIntent
Intent intent = new Intent(this, MyReceiver.class);

// Create the pending intent and wrap our intent
PendingIntent pendingIntent = PendingIntent.getActivity(this, 1, intent, 0);

// Get the alarm manager service and schedule it to go off after 3s
AlarmManager alarmManager =
    (AlarmManager) getSystemService(ALARM_SERVICE);
alarmManager.set(AlarmManager.RTC_WAKEUP, System.currentTimeMillis()
+ (seconds * 1000), pendingIntent);
```



13 – Notificaciones

Més opcions per a notificaciones

En versions a partir de la 4.0 s'han afegit altres visualitzacions per les notificaciones

InboxStyle

Permet mostrar varies lines de missatges:

```
Notification noti = new Notification.Builder()  
    .setContentTitle("5 New mails from " + sender.toString())  
    .setContentText(subject)  
    .setSmallIcon(R.drawable.new_mail)  
    .setLargeIcon(aBitmap)  
    .setStyle(new Notification.InboxStyle()  
        .addLine(str1)  
        .addLine(str2)  
        .setContentTitle("")  
        .setSummaryText("+3 more"))  
    .build();
```



13 – Notificaciones

Més opcions per a notificaciones

BigPictureStyle

Mostra una imatge gran

```
Notification notif = new Notification.Builder(mContext)
    .setContentTitle("New photo from " + sender.toString())
    .setContentText(subject)
    .setSmallIcon(R.drawable.new_post)
    .setLargeIcon(aBitmap)
    .setStyle(new Notification.BigPictureStyle()
        .bigPicture(aBigBitmap))
    .build();
```



13 – Notificaciones

Més opcions per a notificaciones

Progress bar en una notificacio

Es pot mostrar una barra de progres i anar-la actualitzant

mBuilder.setProgress(int maxvalue, int currentValue, boolean determinate)

```
// Sets the progress indicator to a max value, the current completion
// percentage, and "determinate" state
mBuilder.setProgress(100, incr, false);
// Issues the notification
mNotifyManager.notify(0, mBuilder.build());
```

```
// Sets an activity indicator for an operation of indeterminate length
mBuilder.setProgress(0, 0, true);
// Issues the notification
mNotifyManager.notify(0, mBuilder.build());
```



13 – Notificaciones

Més opcions per a notificaciones

Mostrar un custom layout

Podem crear un layout en XML i inflar-lo, mitjançant un **RemoteViews**

```
RemoteViews contentView = new RemoteViews(getPackageName(),  
    R.layout.custom_notification);  
contentView.setImageDrawableResource(R.drawable.ic_launcher);  
contentView.setTextViewText(R.id.title, "Custom notification");  
contentView.setTextViewText(R.id.text, "This is a custom layout");  
notification.contentView = contentView;
```

