**Servizio Notifiche**

Gli utenti Android sono abituati a ricevere notifiche. Si manifestano con una piccola icona che appare nella cosiddetta “Notification Area” e se ne può consultare il contenuto aprendo il “Notification drawer”, una zona “a scomparsa” sul display. Oltre alla semplicità comunicativa che le contraddistingue e alla familiarità dell’utente con questo meccanismo offrono un esempio di segnalazione che esula dall’interfaccia dell’applicazione. Se si osserva la figura (dalla documentazione ufficiale Android) si possono riconoscere i vari elementi che costituiscono una comune notifica. Essi sono:

1. titolo della notifica (*content title*);
2. icona grande (*large icon*);
3. contenuto della notifica (*content text*);
4. informazioni accessorie (*content info*);
5. icona piccola (*small icon*) che di norma appare anche nella barra del display;
6. ora della notifica (*when*) impostata dal programmatore o di default dal sistema.

**La prima notifica**

Visto che le notifiche appaiono in zone del display non gestite dall’applicazione, dovremo interagire con il sistema mediante un apposito servizio: il **NotificationManager**. Nonostante la molteplicità di aspetti che contraddistinguono una notifica, ve ne sono **tre assolutamente obbligatori**:

* l’icona piccola;
* il titolo;
* il contenuto.

Questi saranno configurati, rispettivamente, con i metodi setSmallIcon, setTitle e setContentText.

La creazione della notifica ha seguito due fasi:

* è stata costruita mediante il *Builder* incluso nella classe NotificationCompat. I vari metodi *setter* permetteranno di configurarne i molti aspetti. In questo caso, si è provveduto al minimo indispensabile;
* finita la fase di build, la notifica verrà pubblicata dal NotificationManager mediante invocazione del metodo notify.

**Collegare un’azione alla notifica**

Far apparire notifiche non soddisfa le necessità dell’utente. Egli è abituato a cliccarvi sopra per ottenere una reazione da parte dell’applicazione. Noi, faremo in modo che **il click sulla notifica provochi l’apertura di un’altra Activity** denominata MessageActivity. Ricordiamo comunque che si dovranno seguire due step fondamentali: creare la classe MessageActivity estendendo Activity oltre all’eventuale layout ed inserire un nodo di configurazione del nuovo componente nell’*AndroidManifest.xml*.

L’apertura dell’Activity avverrà mediante Intent ma non sarà attivata subito con il metodo startActivity bensì sarà predisposta per “usi futuri” mediante la classe PendingIntent. Si tratta di una classe che, per così dire, conserva l’Intent e la descrizione dell’azione che esso porta con sé per poterlo attivare successivamente. Ciò che faremo sarà:

* predisporre un normale Intent per l’apertura della MessageActivity;
* creazione del PendingIntent sfruttando l’Intent del punto precedente;
* assegnazione del PendingIntent alla notifica mediante il metodo setContentIntent del NotificationCompat.Builder.