

Corso di Laboratorio di Programmazione per Sistemi Mobile e Tablet

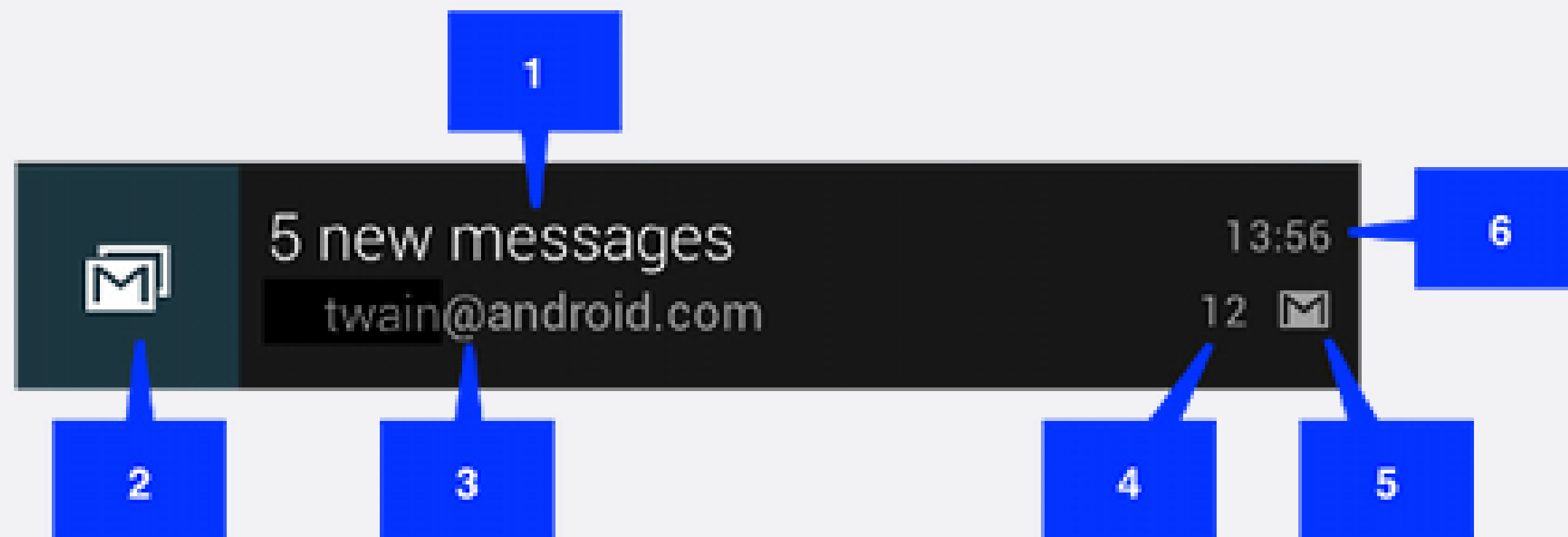
Docente: Dr. Mauro Dragoni

Dipartimento di Ingegneria e Scienze dell'Informazione
Anno Accademico 2021/2022

| in questa lezione...

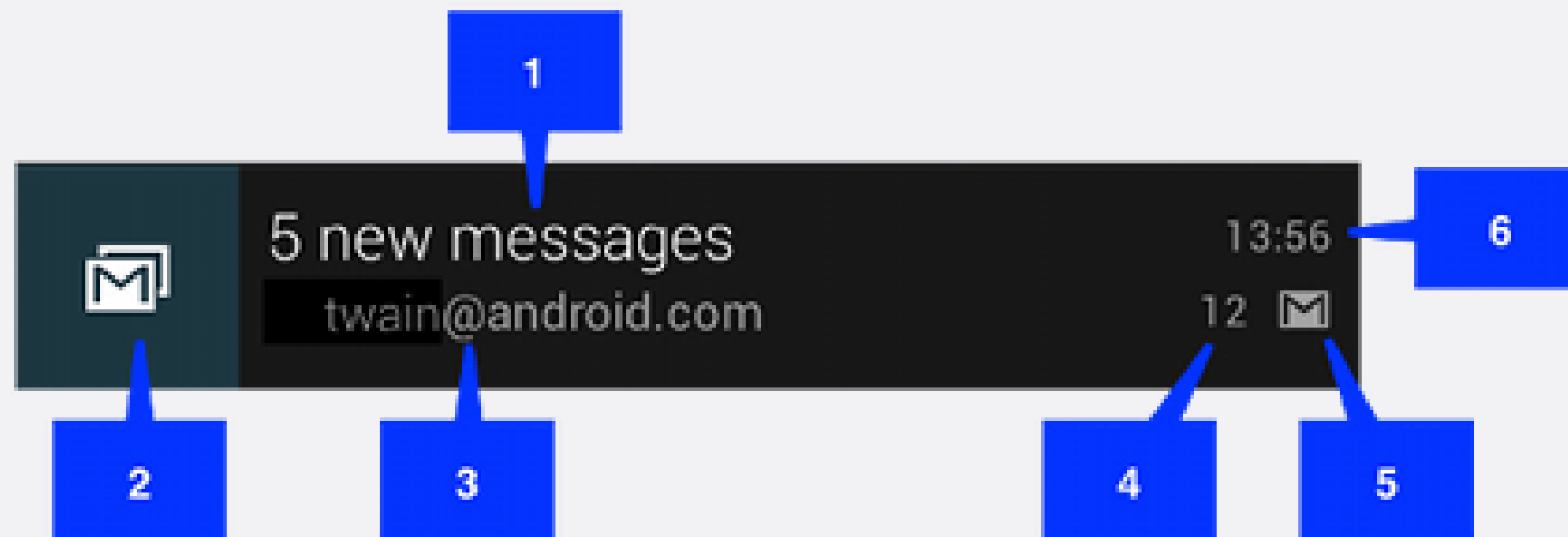
- Notifiche:
 - Gestione delle notifiche
 - Toast
- Real-time messaging

Notifiche: caratteristiche base



- Notifiche:
 - Esempio di interazione con elementi di Android gestiti al di fuori della vostra applicazione.
 - Avviene tramite la gestione della notification area.

Notifiche: caratteristiche base



1. titolo della notifica (*contentTitle*)
2. icona grande (*largeIcon*)
3. contenuto della notifica (*contentText*)
4. informazioni accessorie (*contentInfo*)
5. icona piccola (*smallIcon*)
6. ora della notifica (*when*)

La prima notifica: il NotificationManager

- **NotificationManager**: servizio che permette di interagire con il sistema operativo Android relativamente alla pubblicazione delle notifiche.

```
NotificationManager notificationManager =  
    (NotificationManager) getSystemService (NOTIFICATION_SERVICE) ;
```

- Tre informazioni obbligatorie da impostare:
 1. titolo della notifica (metodo: *setContentTitle*)
 2. contenuto della notifica (metodo: *setContentText*)
 3. icona piccola (metodo: *setSmallIcon*)

La prima notifica: il NotificationManager

- Esempio:

```
NotificationCompat.Builder n = new NotificationCompat.Builder(this)
    .setContentTitle("Arrivato nuovo messaggio!!")
    .setContentText("Autore: LPSMT App")
    .setSmallIcon(android.R.drawable.ic_dialog_email);

NotificationManager notificationManager =
    (NotificationManager) getSystemService(NOTIFICATION_SERVICE);

notificationManager.notify(0, n.build());
```

E se uso API > 26?

- Introduzione del Notification Channel

- Esempio:

```
int importance = NotificationManager.IMPORTANCE_DEFAULT;
NotificationChannel channel =
    new NotificationChannel("myChannelId", "My Channel", importance);
channel.setDescription("Reminders");

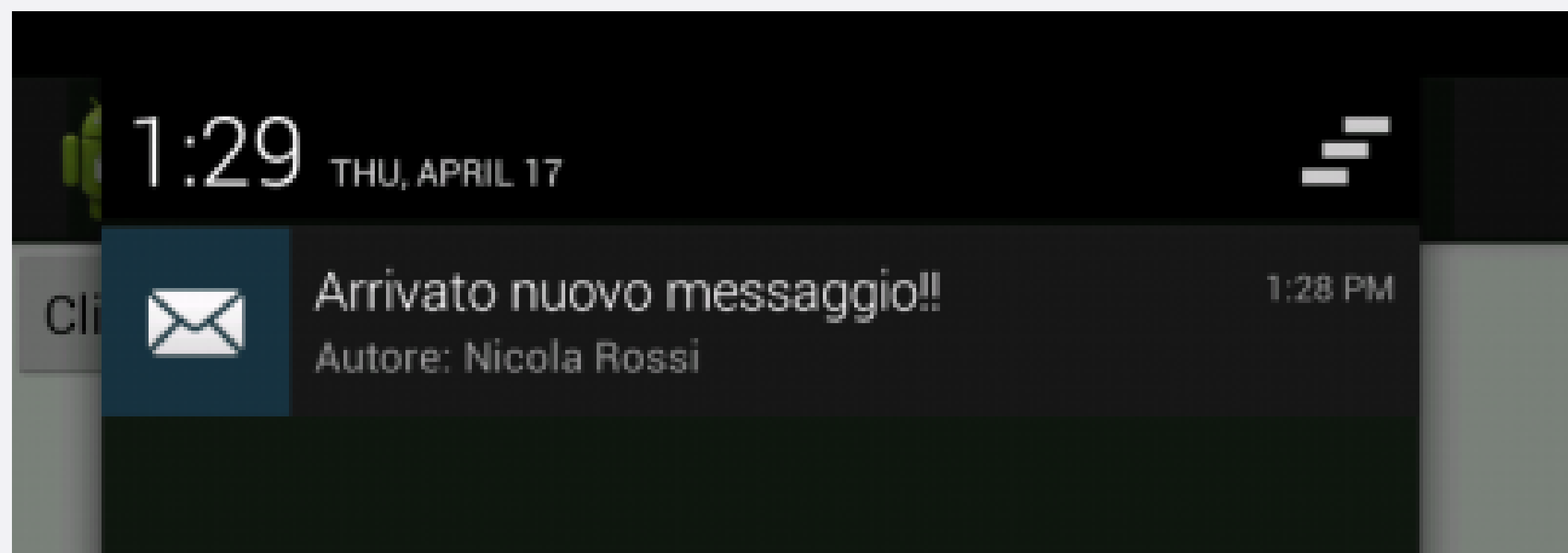
NotificationCompat.Builder n = new NotificationCompat.Builder(this,
    "myChannelId")
    .setContentTitle("New message!!")
    .setContentText("Autore: LPSMT App")
    .setSmallIcon(android.R.drawable.ic_dialog_email);

NotificationManager notificationManager =
    (NotificationManager) getSystemService(NOTIFICATION_SERVICE);

notificationManager.notify(0, n.build());
```

La prima notifica: il NotificationManager

- La creazione della notifica segue quindi due fasi:
 1. costruzione mediante il Builder
 2. pubblicazione della notifica



Avviare una nuova Activity a partire dalla notifica

- L'apertura della `Activity` avviene mediante `Intent`, o meglio mediante la creazione di un `PendingIntent`
- Passi da seguire:
 - preparazione del `Intent`
 - creazione del `PendingIntent`
 - assegnazione del `PendingIntent` alla notifica

Avviare una nuova Activity a partire dalla notifica

- Esempio:

```
Intent i = new Intent(this, MessageActivity.class);
PendingIntent pi = PendingIntent.getActivity(this, 0, i, 0);

NotificationCompat.Builder n = new NotificationCompat.Builder(this)
    . . .
    . . .
    .setContentIntent(pi)
    .setAutoCancel(true);

NotificationManager notificationManager =
    (NotificationManager) getSystemService(NOTIFICATION_SERVICE);

notificationManager.notify(0, n.build());
```

Avviare una nuova Activity a partire dalla notifica

- Esempio con aggiunta dell'avviso sonoro:

```
Intent i = new Intent(this, MessageActivity.class);  
PendingIntent pi = PendingIntent.getActivity(this, 0, i, 0);
```

```
Uri sound =  
RingtoneManager.getDefaultUri(RingtoneManager.TYPE_NOTIFICATION);
```

```
NotificationCompat.Builder n = new NotificationCompat.Builder(this)  
.  
.  
.  
.setContentIntent(pi)  
.setAutoCancel(true);  
.setSound(sound);
```

```
NotificationManager notificationManager =  
    (NotificationManager) getSystemService(NOTIFICATION_SERVICE);
```

```
notificationManager.notify(0, n.build());
```

Toast

- E' la forma di messaggio di dialogo piu' semplice implementabile in Android.

```
Toast.makeText(this, "Ciao a tutti!", Toast.LENGTH_SHORT).show();
```

- Parametri:
 - contesto dell'applicazione
 - testo che si vuole mostrare
 - durata del messaggio (LENGTH_SHORT o LENGTH_LONG)

Push Notifications

- Ci sono diverse modalita' per gestire le push notifications.
- Le due piu' utilizzate sono:
 - Parse Push [[Sito](#)]
 - Google Cloud Messaging [[Sito](#)]

Real-time messaging

- Utilizzare Android per creare e gestire chat sia singole che di gruppo.
- Generalmente ci si appoggia a servizi esterni.
- I principali:
 - PubNub [[Sito](#)]
 - MirrorFly [[Sito](#)]
 - Applozic [[Sito](#)]
 - EnableX [[Sito](#)]