

Android 4.0, Ice Cream Sandwich

Android comes of age with a new, refined design. Simple, beautiful and beyond smart.



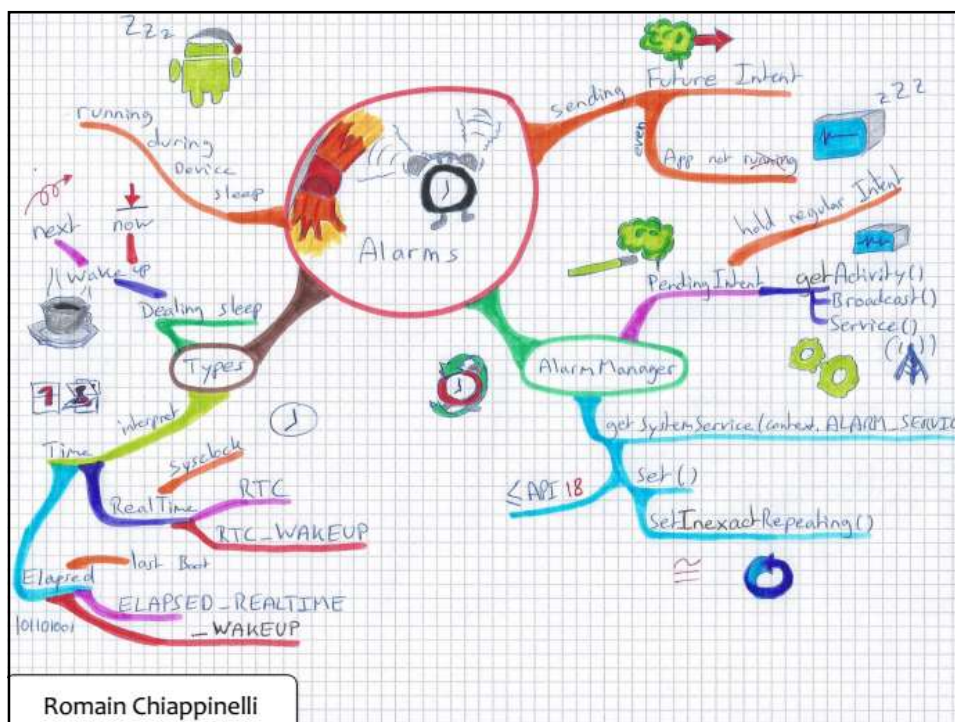
CHAPITRE 8 LES ALARMES



• Mr. MEGHAZI

2019-2020

Cours pour les Master II - GL



Contenu

- Les Alarmes
- Les APIs de l' AlarmManager
- Les Types d'Alarmes
- Exemple d'Application



3

Alarmes (1)

- Mécanismes utilisés pour envoyer des **Intents** à un certain moment(s) **dans le futur**.
- Permettent à une application d'exécuter un code, même si l'application n'est plus en cours d'exécution.

4

Alarmes (2)

- Une fois enregistrés, les Alarmes **demeurent actives** même si l'appareil est endormi.
- On peut même les configurer pour réveiller l'appareil endormi.
- Les alarmes sont **annulées** une fois que l'appareil est **éteint** ou **redémarré**.

5

Exemples d'Alarmes

- MMS - Retry Scheduler
- Settings - Bluetooth Discoverable timeout
- Phone - User Info Cache
- ...

6

AlarmManager

- Créer et gérer les « Alarmes » indirectement, en interagissant avec l'« **AlarmManager** ».
- On se procure une référence à l'« **AlarmManager** », en invoquant la classe « **Context** »:

```
getSystemService(Context.ALARM_SERVICE)
```

7

Créer une Alarme (1)

- //Alarme qui se déclenche une seule fois

```
void set(int type, long triggerAtTime,  
        PendingIntent operation)
```

- **Note** : ce qui suis est valide seulement pour les API <= 18, à partir de l'API 19(KitKat), la livraison des alarmes est inexacte.

8

Créer une Alarme (2)

- //Alarme qui se déclenche +sieurs fois

```
void setRepeating(int type,
                 long triggerAtTime,
                 long interval,
                 PendingIntent operation)
```

9

Créer une Alarme (3)

- //Alarme qui se déclenche +sieurs fois (critère inexact)

```
void setInexactRepeating(int type,
                        long triggerAtTime,
                        long interval,
                        PendingIntent operation)
```

INTERVAL OPTIONS

INTERVAL_FIFTEEN_MINUTES

INTERVAL_HALF_HOUR

INTERVAL_HOUR

INTERVAL_HALF_DAY

INTERVAL_DAY

10

Types d'Alarmes

- Il existe deux degrés de configuration:
 - Comment interpréter le temps ?
 - Quoi faire si l'appareil est endormit quand l'alarme se déclenche ?

11

Interpréter le temps

- **RealTime** – Relatif à l'horloge système
- **Elapsed** – Relatif au temps du dernier démarrage (Boot)

12

Les appareils en veille (endormis)

- Réveiller l'appareil maintenant et livrer l'Intent
- Attendre que l'appareil soit réveiller pour livrer l'Intent

13

Les constantes de type Alarm

- RTC_WAKEUP
- RTC
- ELAPSED_REALTIME
- ELAPSED_REALTIME_WAKEUP

14

PendingIntent

```
PendingIntent getActivity(  
    Context context,  
    int requestCode, Intent intent,  
    int flags, Bundle options)  
  
PendingIntent getBroadcast(  
    Context context,  
    int requestCode,  
    Intent intent, int flags)  
  
PendingIntent getService(  
    Context context,  
    int requestCode,  
    Intent intent, int flags)
```

15

Exemple - AlarmCreateActivity



16

Exemple - AlarmCreateActivity

```

@Override
public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);

    setContentView(R.layout.main);

    // Get the AlarmManager Service
    mAlarmManager = (AlarmManager) getSystemService(ALARM_SERVICE);

    // Create an Intent to broadcast to the AlarmNotificationReceiver
    mNotificationReceiverIntent = new Intent(AlarmCreateActivity.this,
        AlarmNotificationReceiver.class);

    // Create an PendingIntent that holds the NotificationReceiverIntent
    mNotificationReceiverPendingIntent = PendingIntent.getBroadcast(
        AlarmCreateActivity.this, 0, mNotificationReceiverIntent, 0);

    // Create an Intent to broadcast to the AlarmLoggerReceiver
    mLoggerReceiverIntent = new Intent(AlarmCreateActivity.this,
        AlarmLoggerReceiver.class);

    // Create PendingIntent that holds the mLoggerReceiverPendingIntent
    mLoggerReceiverPendingIntent = PendingIntent.getBroadcast(
        AlarmCreateActivity.this, 0, mLoggerReceiverIntent, 0);

```

17

Exemple - AlarmCreateActivity

```

// Set up single alarm Button
final Button singleAlarmButton = (Button) findViewById(R.id.single_alarm_button);
singleAlarmButton.setOnClickListener(new OnClickListener() {
    @Override
    public void onClick(View v) {

        // Set single alarm
        mAlarmManager.set(AlarmManager.RTC_WAKEUP,
            System.currentTimeMillis() + INITIAL_ALARM_DELAY,
            mNotificationReceiverPendingIntent);

        // Set single alarm to fire shortly after previous alarm
        mAlarmManager.set(AlarmManager.RTC_WAKEUP,
            System.currentTimeMillis() + INITIAL_ALARM_DELAY
            + JITTER, mLoggerReceiverPendingIntent);

        // Show Toast message
        Toast.makeText(getApplicationContext(), "Single Alarm Set",
            Toast.LENGTH_LONG).show();
    }
});

```

18

Exemple - AlarmCreateActivity

```
// Set up repeating Alarm Button
final Button repeatingAlarmButton = (Button) findViewById(R.id.repeating_alarm_button);
repeatingAlarmButton.setOnClickListener(new OnClickListener() {

    @Override
    public void onClick(View v) {

        // Set repeating alarm
        mAlarmManager.setRepeating(AlarmManager.ELAPSED_REALTIME,
            SystemClock.elapsedRealtime() + INITIAL_ALARM_DELAY,
            AlarmManager.INTERVAL_FIFTEEN_MINUTES,
            mNotificationReceiverPendingIntent);

        // Set repeating alarm to fire shortly after previous alarm
        mAlarmManager.setRepeating(AlarmManager.ELAPSED_REALTIME,
            SystemClock.elapsedRealtime() + INITIAL_ALARM_DELAY
            + JITTER,
            AlarmManager.INTERVAL_FIFTEEN_MINUTES,
            mLoggerReceiverPendingIntent);

        // Show Toast message
        Toast.makeText(getApplicationContext(), "Repeating Alarm Set",
            Toast.LENGTH_LONG).show();
    }
});
```

19

Exemple - AlarmCreateActivity

```
// Set up inexact repeating alarm Button
final Button inexactRepeatingAlarmButton = (Button) findViewById(R.id.inexact_repeating_alarm_button);
inexactRepeatingAlarmButton.setOnClickListener(new OnClickListener() {

    @Override
    public void onClick(View v) {

        // Set inexact repeating alarm
        mAlarmManager.setInexactRepeating(
            AlarmManager.ELAPSED_REALTIME,
            SystemClock.elapsedRealtime() + INITIAL_ALARM_DELAY,
            AlarmManager.INTERVAL_FIFTEEN_MINUTES,
            mNotificationReceiverPendingIntent);

        // Set inexact repeating alarm to fire shortly after previous alarm
        mAlarmManager.setInexactRepeating(
            AlarmManager.ELAPSED_REALTIME,
            SystemClock.elapsedRealtime() + INITIAL_ALARM_DELAY
            + JITTER,
            AlarmManager.INTERVAL_FIFTEEN_MINUTES,
            mLoggerReceiverPendingIntent);

        Toast.makeText(getApplicationContext(),
            "Inexact Repeating Alarm Set", Toast.LENGTH_LONG)
            .show();
    }
});
```

20

Exemple - AlarmCreateActivity

```
// Set up cancel repeating alarm Button
final Button cancelRepeatingAlarmButton = (Button) findViewById(R.id.cancel_repeating_alarm_button);
cancelRepeatingAlarmButton.setOnClickListener(new OnClickListener() {

    @Override
    public void onClick(View v) {

        // Cancel all alarms using mNotificationReceiverPendingIntent
        mAlarmManager.cancel(mNotificationReceiverPendingIntent);

        // Cancel all alarms using mLoggerReceiverPendingIntent
        mAlarmManager.cancel(mLoggerReceiverPendingIntent);

        // Show Toast message
        Toast.makeText(getApplicationContext(),
            "Repeating Alarms Cancelled", Toast.LENGTH_LONG).show();
    }
});
}
```