# Alarm Service - сигнализация

# Теория

Служба Alarm Service используется для отправки пользователю разовых или повторяющихся сообщений в заданное время.

# Менеджер оповещений AlarmManager

Доступ к службе Alarm Service осуществляется при помощи объекта AlarmManager следующим образом:

AlarmManager manager = (AlarmManager)getSystemService(

Context.ALARM\_SERVICE);

Методы

• cancel() - удаляет все сигнализации

• setTime() - устанавливает системное время

• setTimeZone() - устанавливает временную зону

• set() - задаёт одноразовую сигнализацию

• setExact() - в API 19 (Kitkat) метод set() заменили на новый метод с теми же параметрами

• setRepeating() - задаёт повторяющиеся сигнализации с фиксированным временным интервалом

• setInexactRepeating() - устанавливает повторяющиеся сигнализации без строгого требования к точности периода повторения. Этот метод является предпочтительнее предыдущего для экономии ресурсов системы

# Параметры

Методы set(), setRepeating(), setInexactRepeating() используют следующие параметры:

typeOne - тип используемого времени (системное или всемирное время UTC), который определяется константами

ELAPSED\_REALTIME - запускает ожидающее намерение, основываясь на времени, которое прошло с момента загрузки устройства, но не с момента выхода из режима ожидания. Это время включает любой временной промежуток, в котором устройство находилось в данном режиме. Обратите внимание, что прошедшее время вычисляется на основании того, когда устройство было загружено. Используется системное время

ELAPSED\_REALTIME\_WAKEUP - по прошествии указанного промежутка времени с момента загрузки выводит устройство из спящего режима и запускает ожидающее намерение. Используется системное время

RTC - запускает ожидающее намерение в указанное время, но не выводит устройство из режима ожидания. Используется всемирное время UTC

RTC\_WAKEUP - выводит устройство из режима ожидания для запуска ожидающего намерения в указанное время. Используется всемирное время UTC

• triggerTime - время работы оповещения

• interval - интервал между отправкой повторных сигнализаций в миллисекундах. Также можно использовать константы

INTERVAL\_DAY

INTERVAL\_HALF\_DAY

INTERVAL\_HOUR

INTERVAL\_HALF\_HOUR

INTERVAL\_FIFTEEN\_MINUTES

• operation - объект PendingIntent, определяющий действие, выполняемое при запуске сигнализации. Можно получить через специальные методы:

PendingIntent.getActivities(Context, int, Intent[], int)

PendingIntent.getActivity(Context, int, Intent, int)

PendingIntent.getService(Context, int, Intent, int)

PendingIntent.getBroadcast(Context, int, Intent, int)

# Служба

Для установки сигнализации вам придётся создать собственную службу, наследуясь от базового класса Service (либо через приёмник BroadcastReceiver):

public class AlarmService extends Service {...}

Запуск и управление службой происходит при помощи объекта Intent.

Установка времени для сигнализаций

Для задания времени работы оповещения необходимо установить его время запуска и добавить к нему длительность работы этого оповещения. Например, нам необходимо, чтобы оповещение отрабатывало 5 секунд после запуска:

long time = SystemClock.elapsedRealTime() + 5000;

Также можно использовать объект Calendar. Например, мы хотим, чтобы продолжительность сигнала оповещения была 10 секунд, а период повторения оповещения был один час:

Calendar calendar = Calendar.getInstance();

calendar.setTimeInMillis(System.currentTimeMillis());

calendar.add(Calendar.SECOND, 10);

long time = calendar.getTimeInMillis();

manager.setRepeating(AlarmManager.RTC\_WAKEUP, time,

AlarmManager.INTERVAL\_HOUR, pendingIntent);