



Notifications & Alarms

Programming the Android Platform

Outline

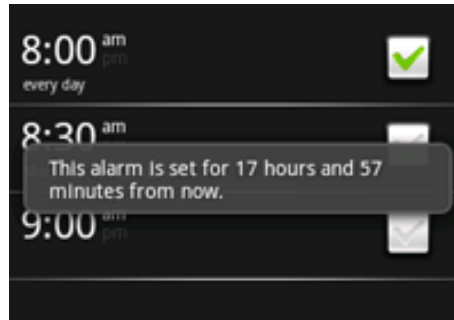
- Notifications
- Alarms

Notification

- Used to notify users of events
- Three general forms of Notifications
 - Toast
 - Dialogs
 - Status Bar Notifications

Toast Notifications

- Transitory messages that pop up on the current window
- Automatically fade into & out of view
- No user interaction or response



Creating Toast Notifications

- Instantiate a Toast object using `makeText()`
 - Parameters - (context, text, duration)
- Can create customized layouts in XML
 - Set them using `Toast.setView()`

Standard Toast Notifications

```
Context context = getApplicationContext();  
CharSequence text = "Message to Display";  
int duration = Toast.LENGTH_SHORT;  
Toast toast = Toast.makeText(context, text, duration);  
// Don't forget to show it! I mess this up every time 😊  
toast.show();
```

custom_toast.xml

```
<LinearLayout ...  
    android:id="@+id/toast_layout_root" android:orientation="horizontal"  
    android:layout_width="fill_parent" android:layout_height="fill_parent"  
    android:padding="10dp" android:background="#7777" >  
    <ImageView  
        android:layout_width="wrap_content" android:layout_height="fill_parent"  
        android:layout_marginRight="10dp"  
        android:src="@drawable/fire_eye_alien" />  
    <TextView  
        android:layout_width="wrap_content" android:layout_height="fill_parent"  
        android:textColor="#FFF" android:text="I'm Watching You!" />  
</LinearLayout>
```

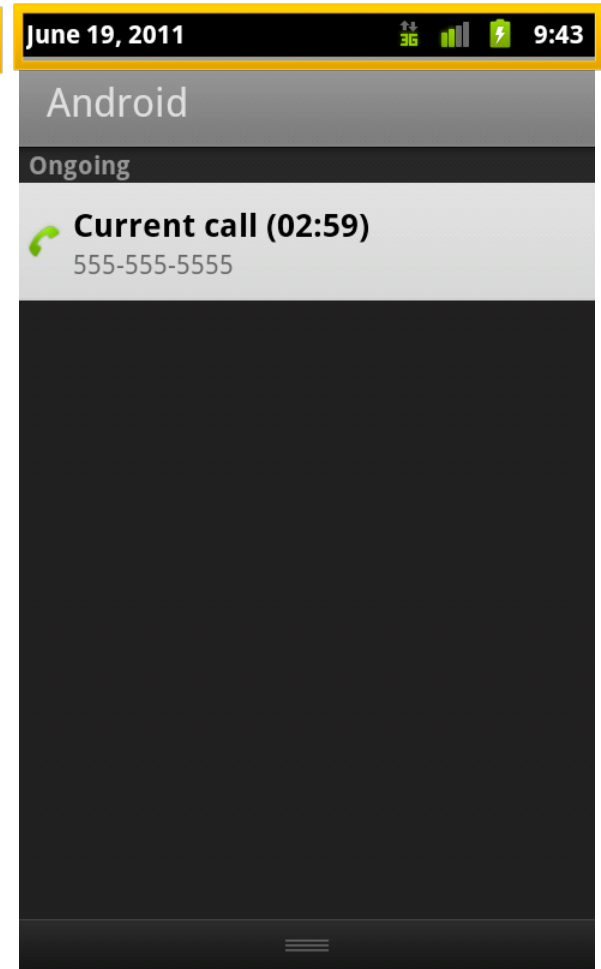
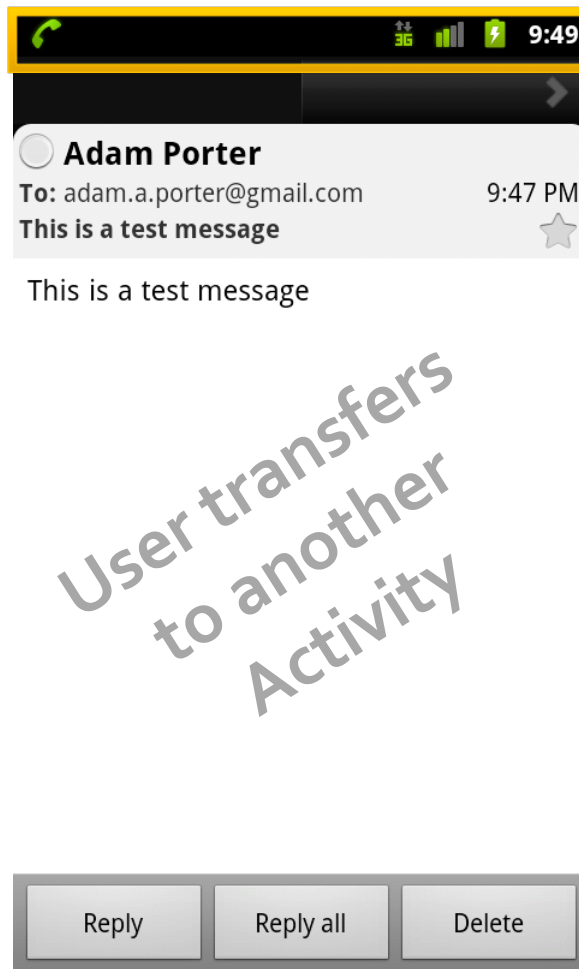
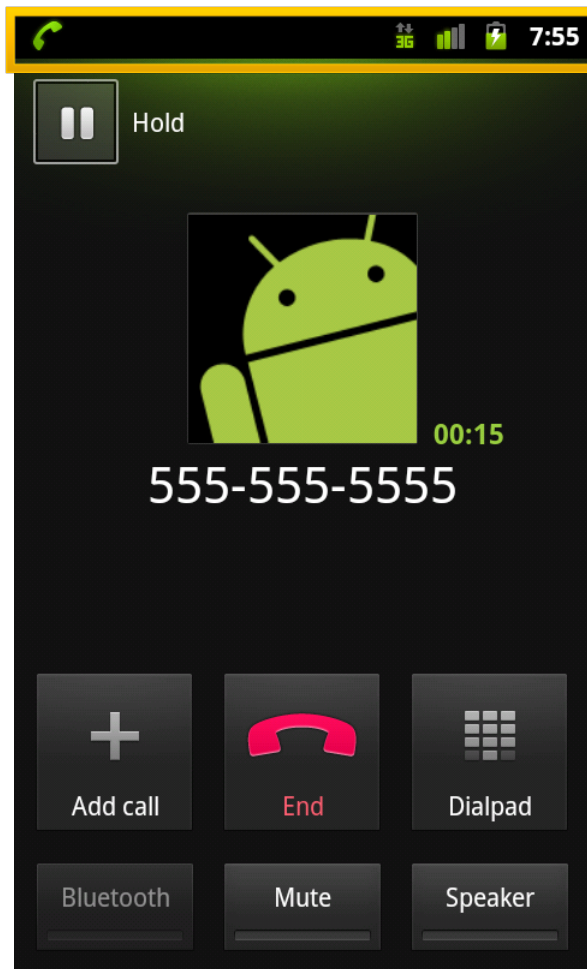
Custom Toast Notifications

```
public class NotificationToastActivity extends Activity {  
    public void onCreate(Bundle savedInstanceState) {  
        ...  
        setContentView(R.layout.main);  
        LayoutInflater inflater = getLayoutInflater();  
        View layout = inflater.inflate(  
            R.layout.custom_toast,  
            (ViewGroup) findViewById(R.id.toast_layout_root));  
        Toast toast = new Toast(getApplicationContext());  
        toast.setGravity(Gravity.CENTER_VERTICAL, 0, 0);  
        toast.setDuration	Toast.LENGTH_LONG);  
        toast.setView(layout);  
        toast.show();  
    }  
}
```


Dialogs

- Created & displayed as part of an Activity
 - Shown & removed using `showDialog()` & `dismissDialog()`
- Can define action to perform on dismissal using `DialogInterface.OnDismissListener` interface
 - `void onDismiss(DialogInterface dialog)`
- See previous UI lesson for examples

Status Bar Notifications



Notification Manager

- System service that manages Notifications
 - e.g., notify & cancel

Creating Notifications

```
public static class MyService extends Service {  
    private static final int NOTIFICATION_ONE = 1;  
    ...  
    public void onStart(Intent intent, int startId) {  
        ...  
        int icon = android.R.drawable.stat_sys_warning;  
        CharSequence tickerText = "Notification";  
        long when = System.currentTimeMillis();  
        Notification notification = new Notification(icon, tickerText, when);  
        ...  
    }  
}
```

Expanded View & Intent

...

```
Context context = getApplicationContext();
```

```
CharSequence contentTitle = "My Notification";
```

```
CharSequence contentText = "You've got notifications!";
```

```
Intent notificationIntent =
```

```
    new Intent(this, NotificationStatusBarActivity.class);
```

```
PendingIntent contentIntent =
```

```
    PendingIntent.getActivity(this, 0, notificationIntent, 0);
```

```
notification.setLatestEventInfo(
```

```
    context, contentTitle, contentText, contentIntent);
```

...

Send Notification

...

```
NotificationManager mNotificationManager = (NotificationManager)
    getSystemService(Context.NOTIFICATION_SERVICE);
mNotificationManager.notify(NOTIFICATION_ONE, notification);
```

...

Adding Sound

- To use the user's default sound
 - `notification.defaults |= Notification.DEFAULT_SOUND;`
- To use a different sound
 - `notification.sound = Uri.parse("file:///sdcard/media/audio/notifications/rooster_2.mp3");`

Adding Vibration

- To use the user's default vibration pattern
 - `notification.defaults |= Notification.DEFAULT_VIBRATE;`
- To use a different vibration pattern
 - `long[] vibrate = { 0, 100, 200, 300 };`
 - `notification.vibrate = vibrate;`

Alarms

- Mechanism for broadcasting Intents at predetermined times
- Used to start Activities at a specific time or perform actions at periodic time intervals
- Once set, Alarms remain active even if target application is inactive or asleep
 - Can optionally be set to wake the device
- Canceled on device shutdown/restart

AlarmManager

■ Example methods

- `void cancel(PendingIntent operation)`
- `void set(int type, long triggerAtTime, PendingIntent operation)`
- `void setRepeating(int type, long triggerAtTime,
long interval, PendingIntent operation)`
- `void setInexactRepeating(int type, long triggerAtTime,
long interval, PendingIntent operation)`

Alarm Types

- RTC_Wakeup
 - Wakes device, fires Intent at the specified clock time
- RTC
 - Doesn't wake device, fires Intent at the specified clock time
- ELAPSED_REALTIME
 - Doesn't wake device, fires Intent at the specified time (interpreted relative to time since last boot)
- ELAPSED_REALTIME_WAKEUP
 - Wakes device, fires Intent at the specified time (interpreted relative to time since last boot)

Alarm Example

```
public void onCreate(Bundle savedInstanceState) {  
    ...  
    alarmSetButton.setOnClickListener(new OnClickListener() {  
        public void onClick(View v) {  
            Intent intent1 = new Intent(AlarmCreateActivity.this, ScoldActivity.class);  
            intent1.addFlags(Intent.FLAG_ACTIVITY_NEW_TASK);  
            PendingIntent sender1 = PendingIntent.getActivity(  
                AlarmCreateActivity.this, 0, intent1,  
                PendingIntent.FLAG_UPDATE_CURRENT);  
            Intent intent2 = new Intent(  
                AlarmCreateActivity.this, AlarmCancelReceiver.class);  
            PendingIntent sender2 = PendingIntent.getBroadcast(  
                AlarmCreateActivity.this, 0, intent2,  
                PendingIntent.FLAG_UPDATE_CURRENT);  
            ...  
        }  
    });  
}
```

Alarm Example (cont.)

```
...
AlarmManager am =
    (AlarmManager) getSystemService(ALARM_SERVICE);
Calendar cal = Calendar.getInstance();
am.setRepeating(
    AlarmManager.RTC_WAKEUP, cal.getTimeInMillis(), 10000, sender1);
am.set(
    AlarmManager.RTC_WAKEUP, cal.getTimeInMillis() + 20000, sender2);
...
```

Alarm Example (cont.)

```
public class ScoldActivity extends Activity {  
    protected void onCreate(Bundle savedInstanceState) {  
        ...  
        Toast.makeText(this, "Are you playing Angry Birds again?",  
                        Toast.LENGTH_LONG).show();  
    }  
}
```

Alarm Example (cont.)

```
public class AlarmCancelReceiver extends BroadcastReceiver {  
    public void onReceive(Context context, Intent intent) {  
        Intent newIntent = new Intent(context, ScoldActivity.class);  
        PendingIntent sender = PendingIntent.getActivity( context, 0,  
                                                         newIntent, PendingIntent.FLAG_UPDATE_CURRENT);  
        final AlarmManager am = (AlarmManager)  
            context.getSystemService(Context.ALARM_SERVICE);  
        am.cancel(sender);  
    }  
}
```

Source Code Examples

- NotificationStatusBar
- NotificationStatusBarWithCustomView
- NotificationToast
- AlarmCreateActivity