

PROGRAMMING HANDHELD SYSTEMS

ADAM PORTER

USER NOTIFICATION

TODAY'S TOPICS

TOAST

NOTIFICATION  AREA NOTIFICATIONS

USER NOTIFICATIONS

MESSAGES PROVIDED TO THE USER OUTSIDE
OF THE NORMAL UI

USER NOTIFICATIONS

THESE INCLUDE MESSAGES AIMED AT

USER FEEDBACK 

TOASTS

DIALOGS

EVENT NOTIFICATION 

NOTIFICATION AREA NOTIFICATIONS

TOAST

TRANSITORY MESSAGES THAT POP UP ON THE
CURRENT WINDOW

E.G., TO INFORM USER THAT AN OPERATION HAS
COMPLETED SUCCESSFULLY

AUTOMATICALLY FADE INTO & OUT OF VIEW

NO USER INTERACTION OR RESPONSE

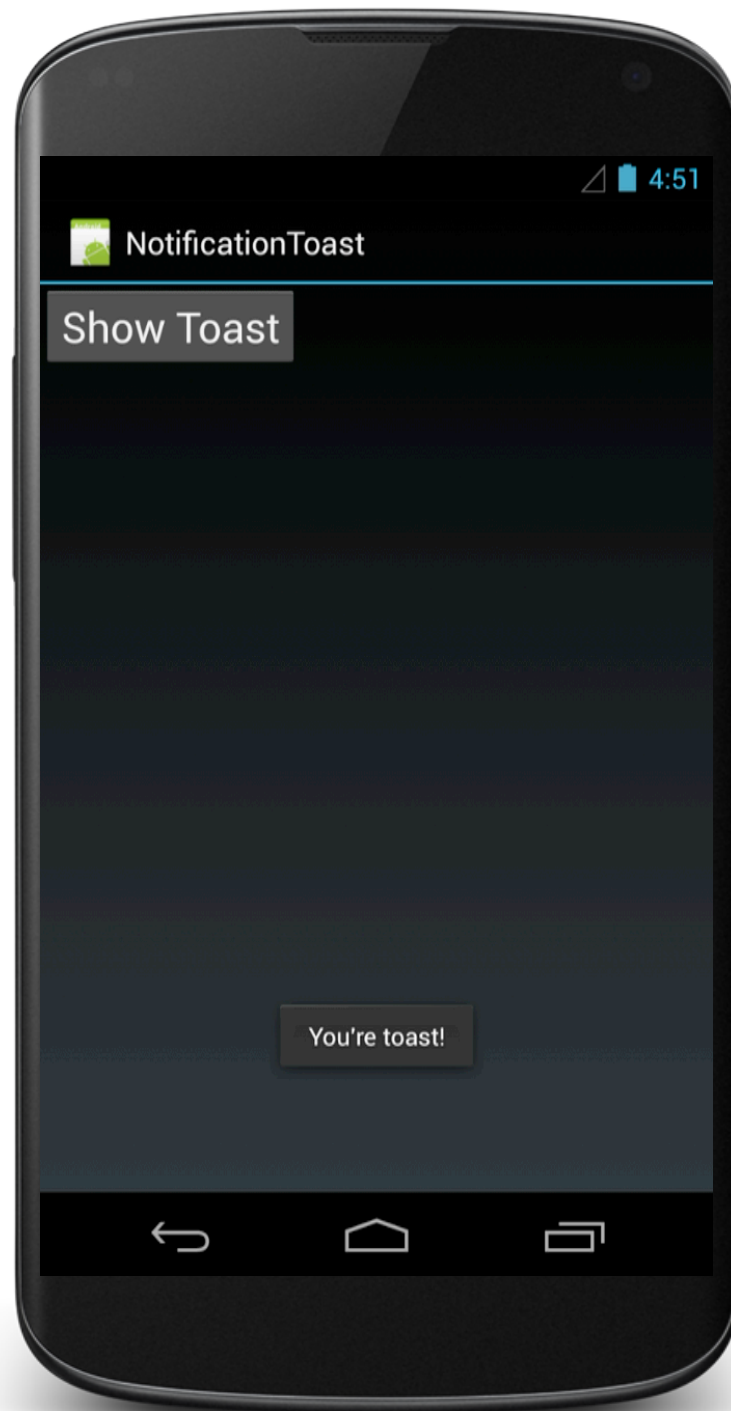
CREATING TOAST NOTIFICATIONS

INSTANTIATE A TOAST OBJECT BY CALLING 

```
Toast.makeText(context, text, duration)
```

SHOW TOAST BY CALLING

```
Toast.show()
```

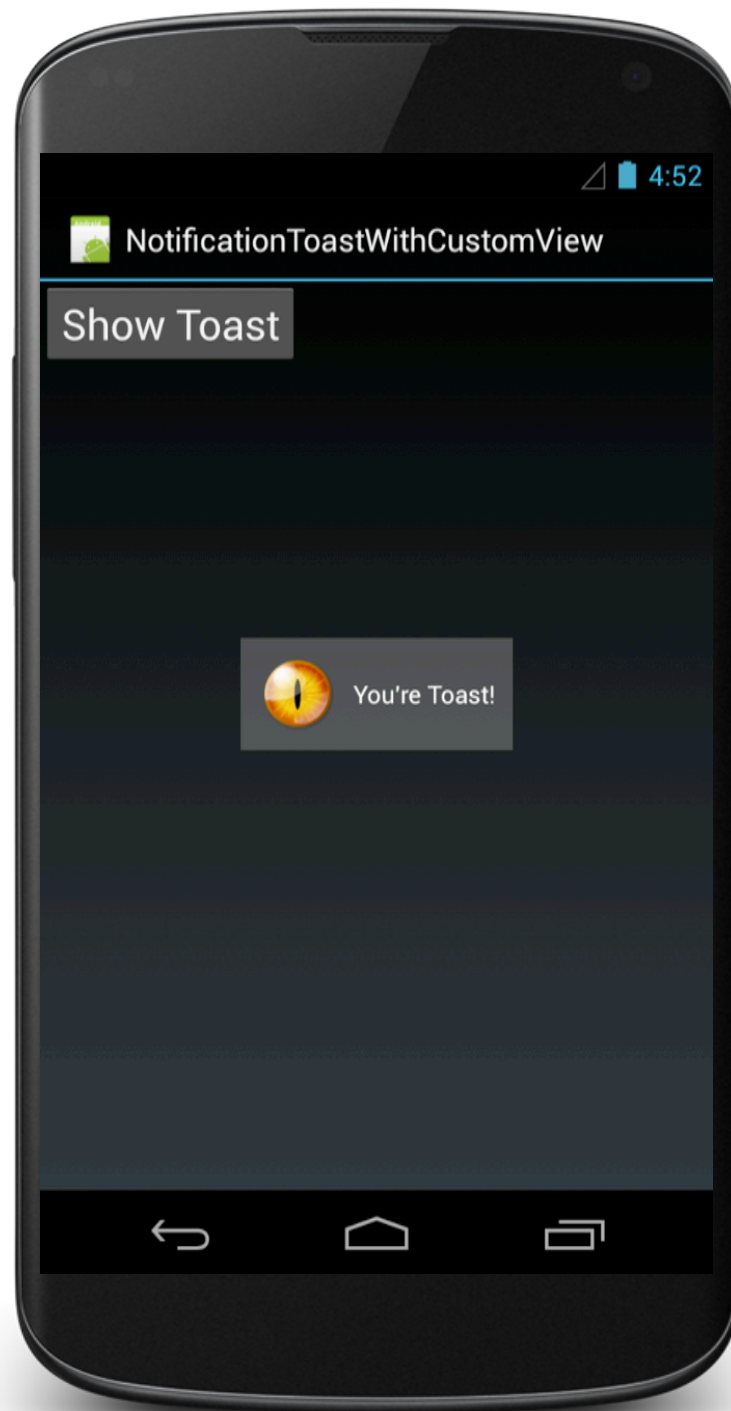
NOTIFICATION TOAST

```
@Override  
public void onClick(View v) {  
    Toast.makeText(getApplicationContext(), "You're toast!", Toast.LENGTH_LONG).show();  
}
```

TOAST WITH CUSTOM VIEWS

CAN CREATE CUSTOM LAYOUTS IN XML

SET THEM USING `Toast.setView()`



NOTIFICATIONTOASTWITHCUSTOMVIEW

```
@Override
public void onClick(View v) {

    Toast toast = new Toast(getApplicationContext());

    toast.setGravity(Gravity.CENTER_VERTICAL, 0, 0);
    toast.setDuration	Toast.LENGTH_LONG);

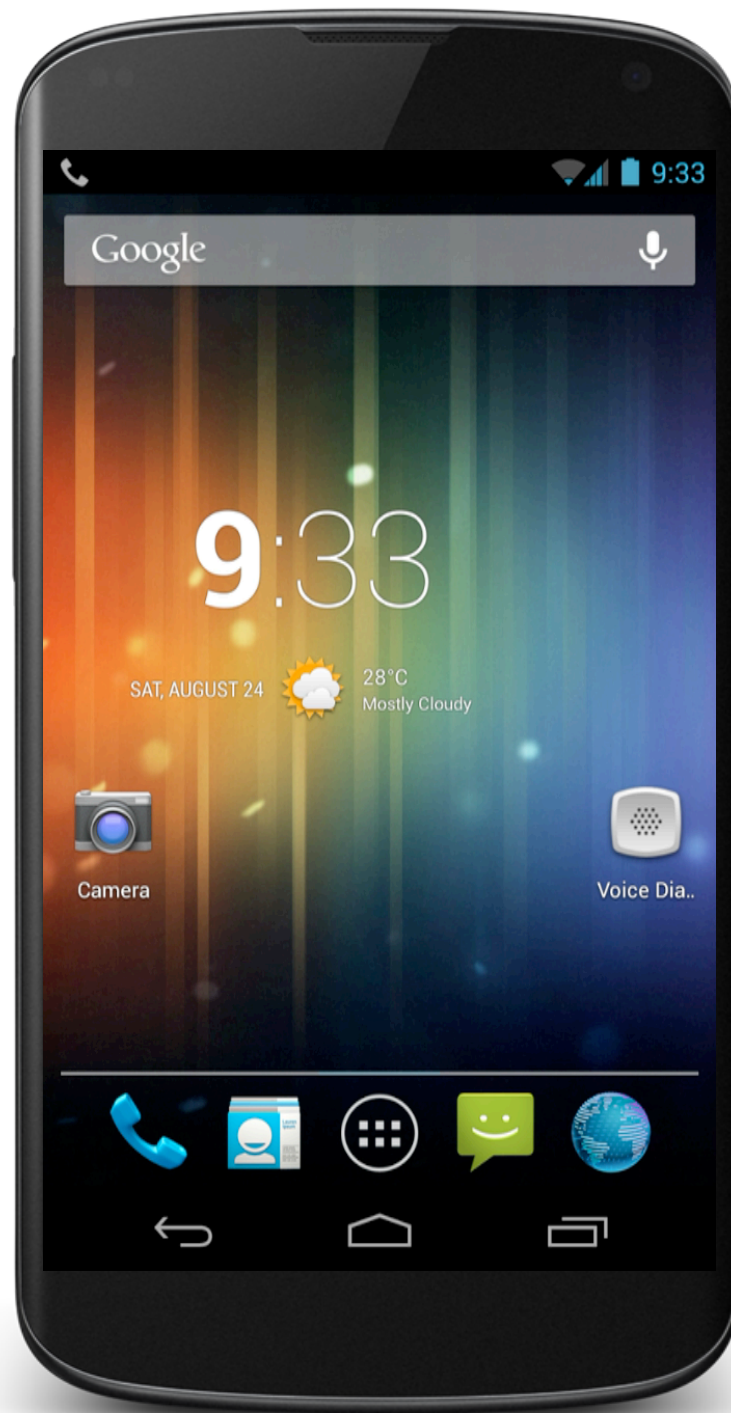
    toast.setView(getLayoutInflater().inflate(R.layout.custom_toast,null));

    toast.show();
}
```

NOTIFICATION AREA NOTIFICATIONS

ANDROID PROVIDES THE NOTIFICATION
AREA FOR ALERTING USERS ABOUT
EVENTS

IT ALSO PROVIDES A NOTIFICATION
DRAWER THAT USERS CAN PULL DOWN
TO SEE MORE DETAILED INFORMATION
ABOUT NOTIFICATIONS



NOTIFICATION ARCHITECTURE

NOTIFICATION

TITLE, DETAIL, SMALL ICON

NOTIFICATION AREA

TICKER TEXT, SMALL ICON

NOTIFICATION DRAWER

VIEW

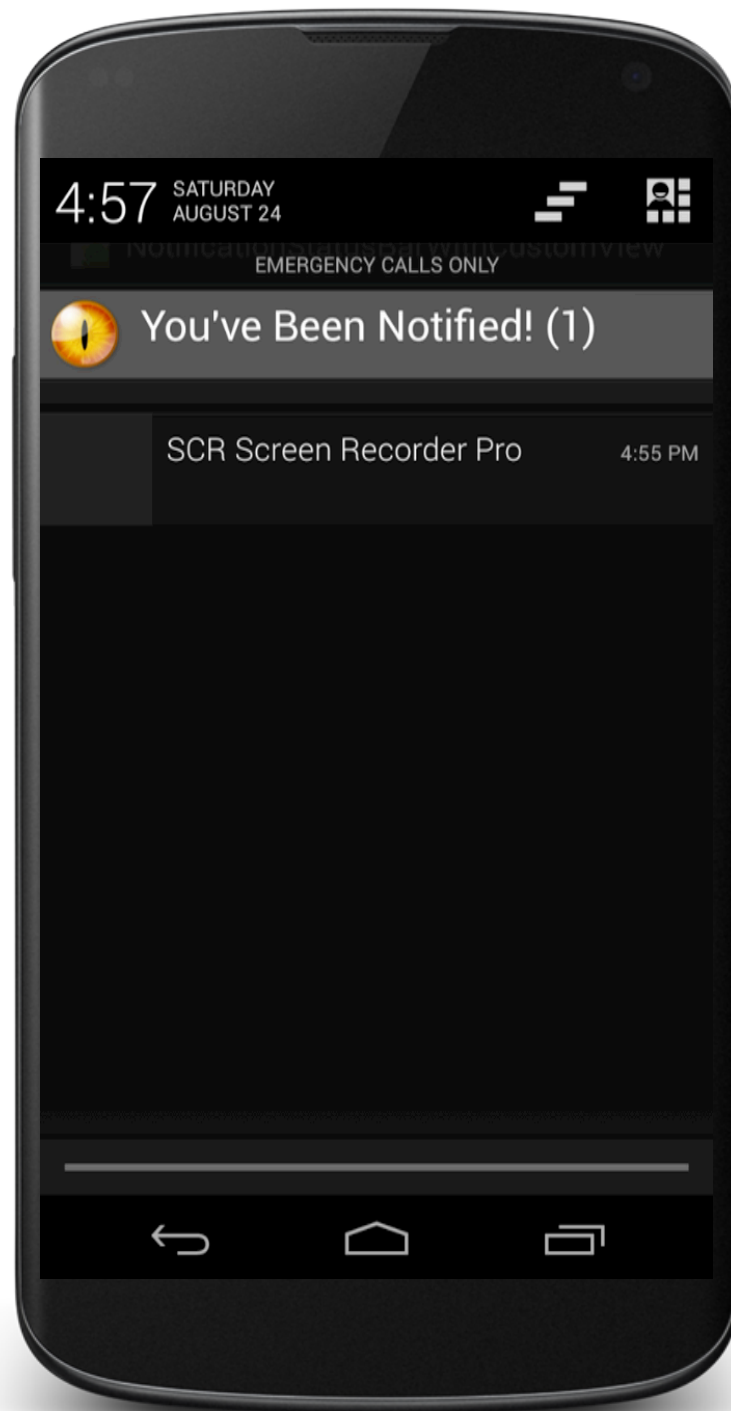
ACTION

NOTIFICATION MANAGER

SYSTEM SERVICE THAT MANAGES
NOTIFICATIONS

SEND & CANCEL NOTIFICATION





NOTIFICATIONSTATUSBARWITHCUSTOMVIEW

```
mNotificationIntent = new Intent(getApplicationContext(),  
    NotificationSubActivityActivity.class);  
mContentIntent = PendingIntent.getActivity(getApplicationContext(), 0,  
    mNotificationIntent, Intent.FLAG_ACTIVITY_NEW_TASK);  
  
final Button button = (Button) findViewById(R.id.button1);  
button.setOnClickListener(new OnClickListener() {  
  
    @Override  
    public void onClick(View v) {  
  
        // Define the Notification's expanded message and Intent:  
  
        mContentView.setTextViewText(R.id.text, contentText + "("  
            + ++mNotificationCount + ")");
```

NOTIFICATIONSTATUSBARWITHCUSTOMVIEW

```
// Build the Notification
```

```
Notification.Builder notificationBuilder = new Notification.Builder(  
    getApplicationContext()  
).setTicker(tickerText)  
.setSmallIcon(android.R.drawable.stat_sys_warning)  
.setAutoCancel(true)  
.setContentIntent(mContentIntent)  
.setSound(soundURI)  
.setVibrate(mVibratePattern)  
.setContent(mContentView);
```

```
// Pass the Notification to the NotificationManager:
```

```
NotificationManager mNotificationManager = (NotificationManager) getSystemService(Context.NOTIFICATION_SERVICE);  
mNotificationManager.notify(MY_NOTIFICATION_ID,  
    notificationBuilder.build());
```

```
    }  
});
```

NOTIFICATIONTOASTWITHCUSTOMVIEW

```
@Override
public void onClick(View v) {

    Toast toast = new Toast(getApplicationContext());

    toast.setGravity(Gravity.CENTER_VERTICAL, 0, 0);
    toast.setDuration	Toast.LENGTH_LONG);

    toast.setView(getLayoutInflater().inflate(R.layout.custom_toast,null));

    toast.show();
}
```


NEXT TIME

THE BROADCASTRECEIVER CLASS