The Android platform is designed to operate on a variety of different device types, screen sizes, and screen resolutions. To assist developers in meeting this challenge, Android provides a rich toolkit of user interface components to utilize and customize to the needs of their specific application. Android also relies very heavily on an extensible XML framework and set resource qualifiers to create liquid layouts that can adapt to these environmental changes. In this chapter, we take a look at some practical ways to shape this framework to fit your specific development needs.

2–1. Customizing the Window

Problem
The default window elements are not satisfactory for your application.

Solution
(API Level 1)
Customize the window attributes and features using themes and the WindowManager. Without any customization, an Activity in an Android application will load with the default system theme, looking something like Figure 2–1.

The window color will be black, with a title bar (often grey) at the top of the Activity. The status bar is visible above everything, with a slight shadow effect underneath it. These are all customizable aspects of the application that are controlled by the Window, and can be set for the entire application or for specific Activities.
CHAPTER 2: User Interface Recipes

How It Works

Customize Window Attributes with a Theme

A Theme in Android is a type of appearance style that is applicable to an entire application or Activity. There are two choices when applying a theme: use a system theme or create a custom one. In either case, a theme is applied in the AndroidManifest.xml file as shown in Listing 2–1.

Listing 2–1. AndroidManifest.xml

```xml
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"…>
  <!--Apply to the application tag for a global theme -->
  <application android:theme="THEME_NAME"…>
    <!--Apply to the activity tag for an individual theme -->
    <activity android:name=".Activity" android:theme="THEME_NAME"…>
      <intent-filter>
        ...
      </intent-filter>
    </activity>
  </application>
</manifest>
```