Material Design for Android

Material design is a comprehensive guide for visual, motion, and interaction design across platforms and devices. Android now includes support for material design apps. To use material design in your Android apps, follow the guidelines defined in the [material design specification](https://material.io/guidelines/) and use the new components and functionality available in Android 5.0 (API level 21) and above.

Android provides the following elements for you to build material design apps:

* A new theme
* New widgets for complex views
* New APIs for custom shadows and animations

For more information about implementing material design on Android, see [Creating Apps with Material Design](https://developer.android.com/training/material/index.html).

Material Theme

The material theme provides a new style for your app, system widgets that let you set their color palette, and default animations for touch feedback and activity transitions.

Using the Material Theme

The new material theme provides:

* System widgets that let you set their color palette
* Touch feedback animations for the system widgets
* Activity transition animations

You can customize the look of the material theme according to your brand identity with a color palette you control. You can tint the action bar and the status bar using theme attributes, as shown in [Figure 3](https://developer.android.com/training/material/theme.html#fig3).

The system widgets have a new design and touch feedback animations. You can customize the color palette, the touch feedback animations, and the activity transitions for your app.

The material theme is defined as:

* @android:style/Theme.Material (dark version)
* @android:style/Theme.Material.Light (light version)
* @android:style/Theme.Material.Light.DarkActionBar

## Apply the Material Theme

To apply the material theme in your app, specify a style that inherits from android:Theme.Material:

<!-- res/values/styles.xml -->  
<resources>  
  <!-- your theme inherits from the material theme -->  
  <style name="AppTheme" parent="android:Theme.Material">  
    <!-- theme customizations -->  
  </style>  
</resources>

## Customize Your Animations

Android 5.0 (API level 21) includes new APIs to create custom animations in your app. For example, you can enable activity transitions and define an exit transition inside an activity:

public class MyActivity extends Activity {  
  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        // enable transitions  
        getWindow().requestFeature(Window.FEATURE\_CONTENT\_TRANSITIONS);  
        setContentView(R.layout.activity\_my);  
    }  
  
    public void onSomeButtonClicked(View view) {  
        getWindow().setExitTransition(new Explode());  
        Intent intent = new Intent(this, MyOtherActivity.class);  
        startActivity(intent,  
                      ActivityOptions  
                          .makeSceneTransitionAnimation(this).toBundle());  
    }  
}