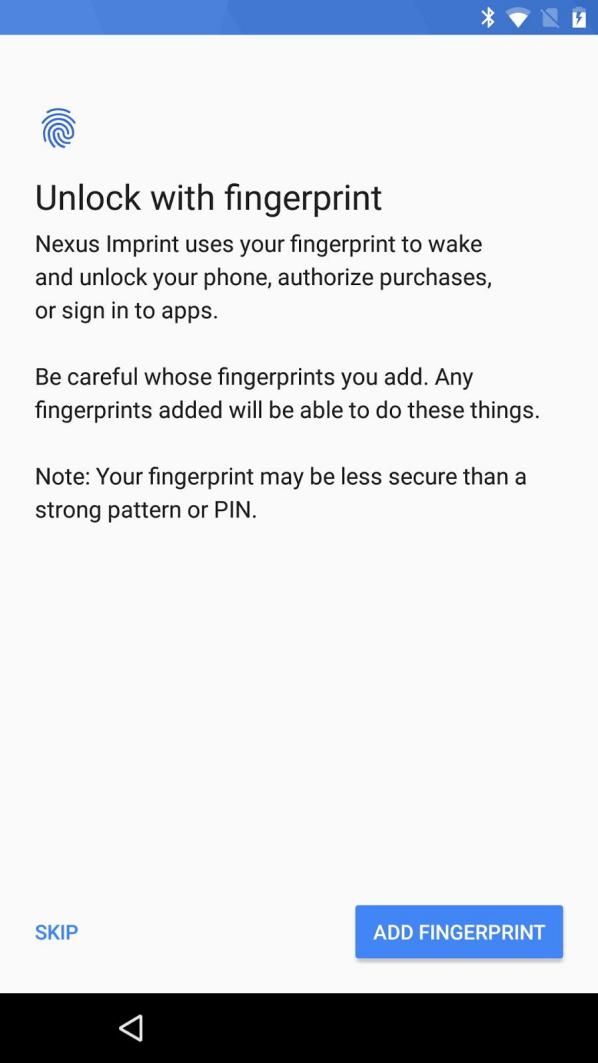
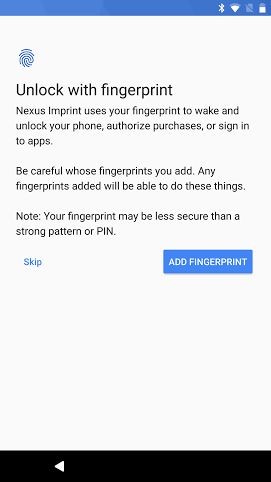
Setup Wizard OEM Guideline

*This doc is a brief guide for Android O Last updated: July 7, 2017*

# What’s new

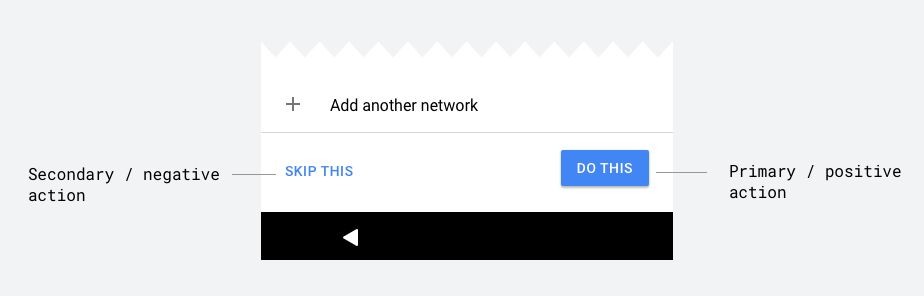
## The Android O release introduces a new UI pattern - sticky footer in setup wizard. It anchors actions at bottom across screens and keeps them always accessible to optimize usability on mobile devices.

Android O版本引入了一个新的UI模式——安装向导中的粘性页脚。它在屏幕的底部锚定动作，并使它们始终能够在移动设备上优化可用性。



Pre-O: inline actions Android O: actions in sticky footer

Updating to sticky footer



For most common use cases of the button footer, you can use the ButtonFooterMixin. For example:

final ButtonFooterMixin buttonFooterMixin = layout.getMixin(ButtonFooterMixin.class);

buttonFooterMixin.addButton("Skip", R.style.SuwGlifButton\_Secondary)

.setOnClickListener(...); buttonFooterMixin.addSpace();

buttonFooterMixin.addButton("Next", R.style.SuwGlifButton\_Primary)

.setOnClickListener(...);

The borderless button on the left is the *secondary* button, represented by

左边的无边界按钮是二级按钮，由

@style/SuwGlifButton.Secondary.

## The raised button on the right is the *primary* button, represented by

右边的凸起按钮是主按钮，由

@style/SuwGlifButton.Primary.

Custom sticky footer

## If you have a fancier footer that ButtonFooterMixin cannot provide, you can add a custom footer with your layout, using the [<aapt:attr>](https://developer.android.com/guide/topics/resources/complex-xml-resources.html) tag.

如果你有一个更漂亮的页脚，那么你可以在你的布局中添加一个定制的页脚，使用<aapt:attr>标签。

<com.android.setupwizardlib.GlifLayout [xmlns:aapt="http://schemas.android.com/aapt"](http://schemas.android.com/aapt) [xmlns:android="http://schemas.android.com/apk/res/android"](http://schemas.android.com/apk/res/android) [xmlns:app="http://schemas.android.com/apk/res-auto"](http://schemas.android.com/apk/res-auto) android:icon="@drawable/my\_icon">

<aapt:attr name="suwFooter">

<LinearLayout style="@style/SuwGlifButtonBar"

android:layout\_width="match\_parent" android:layout\_height="wrap\_content">

<Button

android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" style="@style/SuwGlifButton.Secondary" android:text="@string/secondary\_button\_label" />

<View

android:layout\_width="16dp" android:layout\_height="match\_parent" android:layout\_weight="1" android:background="@drawable/fancy\_divider />

<Button

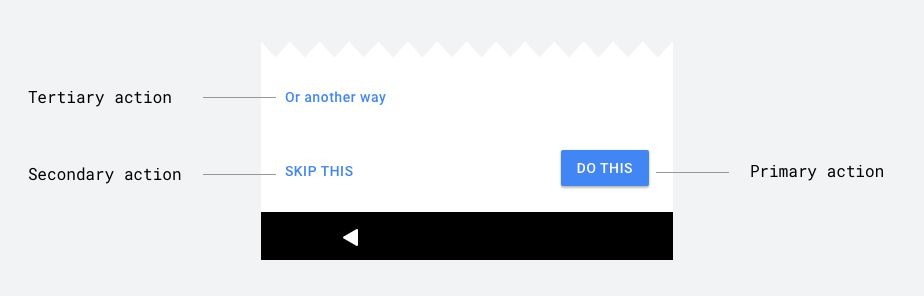
android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" style="@style/SuwGlifButton.Primary" android:text="@string/primary\_button\_label" />

</LinearLayout>

</aapt:attr>

</com.android.setupwizardlib.GlifLayout>

Tertiary (link) buttons



There is also a third type of button, @style/SuwGlifButton.Tertiary, which is not used with the button footer, but rather used in the content area directly. To use that simply apply the style on a button.

(Yes, a Button class, even though it might look like a regular TextView visually. This is to make sure TalkBack announces it as a button)

还有第三种类型的按钮，@style/SuwGlifButton。第三级，它不与按钮页脚一起使用，而是直接在内容区域中使用。要使用它，只需在按钮上应用样式。

（是的，一个按钮类，即使它看起来像一个普通的TextView。这是为了确保TalkBack将其作为一个按钮来宣布）