

Density & resolution

Screen pixel density and resolution vary depending on the platform.

CONTENTS

[Pixel density](#)

[Density independence](#)

[Pixel density on Android](#)

[Pixel density on iOS](#)

[Pixel density on the web](#)

Pixel density

The number of pixels that fit into an inch is referred to as pixel density.

Screen density variations



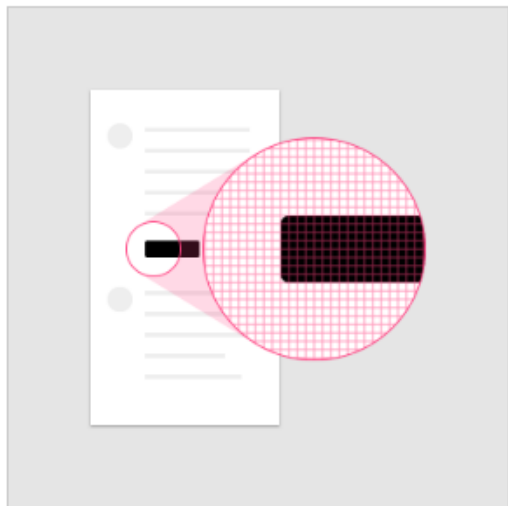
High-density screens have more pixels per inch than low-density ones. As a result, UI elements of the same pixel dimensions appear larger on low-density screens, and smaller on high-density screens.

Calculating pixel density

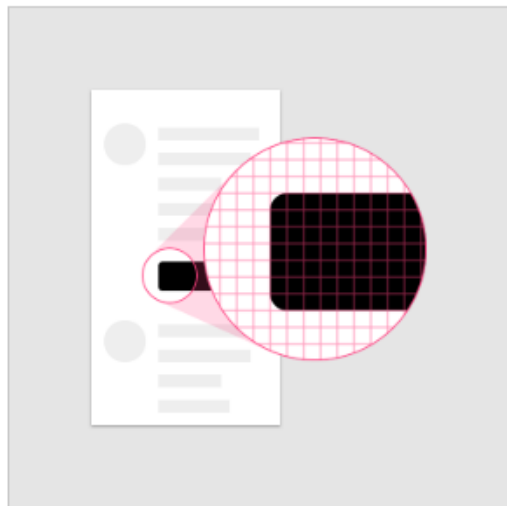


To calculate screen density, you can use this equation:

Screen density = Screen width (or height) in pixels / Screen width (or height) in inches



High-density display



Lower density display

Density independence

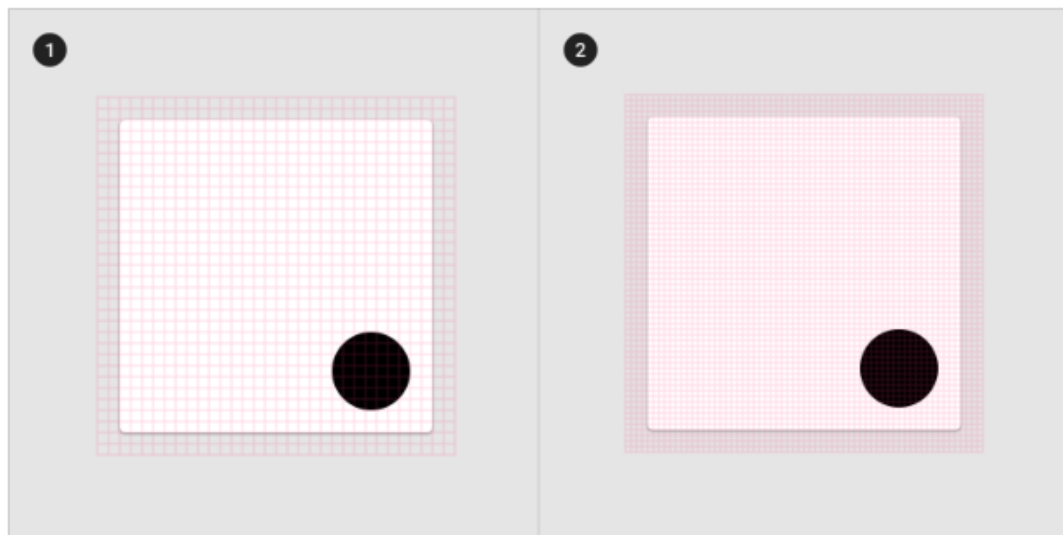
Density independence refers to the uniform display of UI elements on screens with different densities.

Density-independent pixels



Density-independent pixels, written as dp (pronounced “dips”), are flexible units that scale to have uniform dimensions on any screen. They provide a flexible way to accommodate a design across platforms.

Material UIs use density-independent pixels to display elements consistently on screens with different densities.



1. Low-density screen displayed with density independence

2. High-density screen displayed with density independence

Pixel density on Android

When developing an Android app, use dp to display elements uniformly on screens with different densities.

Dps and screen density



A dp is equal to one physical pixel on a screen with a density of 160.

To calculate dp:

$$\text{dp} = (\text{width in pixels} * 160) / \text{screen density}$$

Screen physical width	Screen density	Screen width in pixels	Screen width in dps
1.5 in	120	180 px	240 dp
1.5 in	160	240 px	
1.5 in	240	360 px	

Scalable pixels (sp)



Scalable pixels (sp) serve the same function as density-independent pixels (dp), but for fonts. The default value of an sp is the same as the default value for a dp.

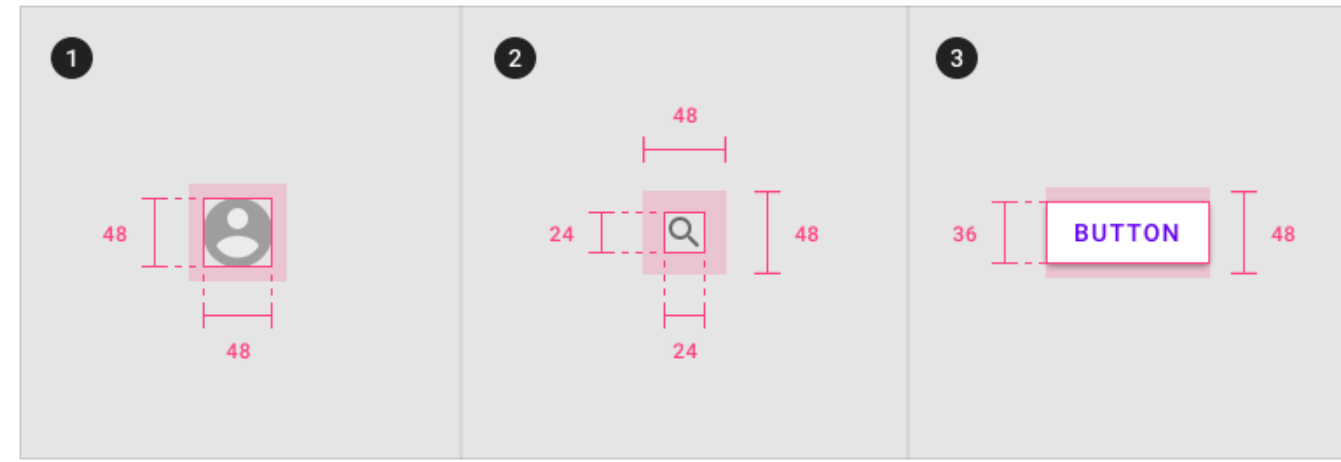
The primary difference between an sp and a dp is that sp's preserve a user's font settings. Users who have larger text settings for accessibility will see font sizes match their text size preferences.

- Material System
 - Introduction
 - Material studies
- Material Foundation
 - Foundation overview
 - Environment
 - Layout
 - Understanding layout
 - Density & resolution
 - Responsive layout grid
 - Spacing methods
 - Baseline
 - Spacing
 - Containers and aspect ratios
 - Touch and click targets**

Touch and click targets

Touch target specs

Touch targets apply to any device that receives both touch and non-touch input. To balance information density and usability, touch targets should be at least 48 x 48 dp with at least 8dp of space between them.



Touch target minimum of 48 x 48 dp

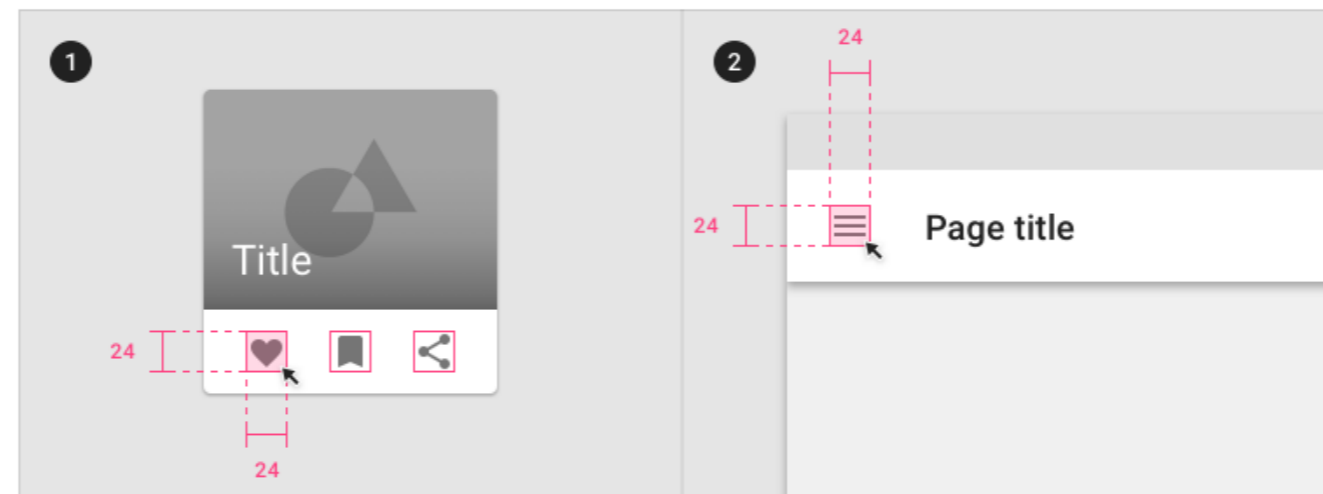
Click targets

On non-touch UIs, click targets should be at least 24 x 24 dp with at least 8dp of space between them.

Device Metrics
A comprehensive resource for sizing, resolution, and more across multiple...
Related Link ↗

Click targets

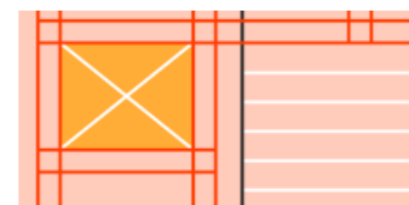
On non-touch-UIs, click targets should be at least 24 x 24 dp with at least 8dp of space between them..



Click target minimum of 24 x 24 dp

Device Metrics
A comprehensive resource for sizing, resolution, and more across multiple...
Related Link [↗](#)

Up next



LAYOUT



COMPONENTS



COMPONENTS



Material System

Introduction

Material studies

Material Foundation

Foundation overview

Environment

Layout

Understanding layout

Density & resolution

Responsive layout grid

Spacing methods

Component behavior

Positioning terminology

Component width

Responsive patterns

Density

Navigation

Color

Typography

Iconography

Shape

Positioning terminology

UI elements behave in different ways at each breakpoint range.

Descriptors



The position of UI elements, components, and surfaces in the grid can be described using the following terms:

Descriptor	Definition
Above, below	The y position of an element
In front of, behind	The z position of an element
Left, right, centered	The x position of an element
Top, bottom	The y position of an element relative to a container or screen edge
Vertically centered	The x and y position of an element are centered relative to a container or screen edges
Sticky	An element that scrolls with the UI and locks at a certain point in the scroll position
Floating	A fixed element positioned in front of scrolling content

Component width

Component width can remain the same across screen sizes, or it can change depending on the layout. Component width can be either:

- Fixed
- Fluid

Because fixed width elements retain their width during layout changes, their placement can change to accommodate new layouts. Their placement can be either:

- Pushed
- Overlaid