TAPFLARE / RESEARCH & PRACTICAL GUIDANCE

Website Mockup Screen Widths: A Source-Labelled Sheet

> Tapflare is a flat-rate monthly subscription for graphic design and website development. A dedicated project manager assigns each request to senior designers and developers, so companies get steady creative and web output without hiring.

Published by Tapflare · 2026-11-18 · website mockup screen widths · figma frame sizes for web design · responsive breakpoints · mobile mockup width 375 or 390 · desktop mockup width · tailwind breakpoints · bootstrap breakpoints · statcounter screen resolution · website design brief

Original article: https://tapflare.com/articles/website-mockup-screen-widths-frame-sheet



In this article

1. Executive Summary
2. Introduction and Background
3. Sheet A: Framework Defaults
4. Sheet B: Design-Tool and Site-Builder Presets
5. Sheet C: Platform Window Classes
6. Sheet D: Analytics Shares
7. Data Analysis and Evidence
8. How to Choose Widths for Your Audience
9. The Brief Clause and Acceptance Test
10. When the Need Is Different
11. Implications and Future Directions
12. Frequently Asked Questions (FAQs)
13. Conclusion

Executive Summary

There is no standard width for a website mockup frame. Every number in circulation was published by someone for a different job, and the sources do not agree. Tailwind CSS ships five default breakpoints starting at **640 px** (tailwindcss.com), Bootstrap 5.3 starts its first media-query tier at **576 px** (getbootstrap.com), Figma Sites presets are Desktop **1280**, Tablet **800** and Mobile **375** (help.figma.com), and Android's window size classes switch at **600, 840, 1200 and 1600 dp** (developer.android.com). These figures describe CSS framework defaults, design-tool presets, and platform window classes. None of them is a measurement of what visitors use, and none of them is a requirement.

Analytics shares are a fourth kind of number, and they answer a different question. StatCounter's October 2026 desktop chart lists 1920x1080 at **27.82%**, 1536x864 at **9.13%**, 800x600 at **8.85%** and 1366x768 at **7.48%** (gs.statcounter.com), while its mobile chart is led by 414x896 at **12.67%** and 360x800 at **8.57%** (gs.statcounter.com). StatCounter's own pages do not say whether "screen resolution" means the physical screen or the browser viewport, so these shares are a rough cross-check and not a design target (gs.statcounter.com). Its figures can also be revised for 45 days after first publication (gs.statcounter.com).

The practical consequence is that a brief should name the widths itself instead of saying "standard sizes". This report proposes a clause: list three or four frame widths, name one as the approval width, and state that every width in between must not break. The accessibility anchor for "in between" is WCAG 2.2 Success Criterion 1.4.10 (Reflow), which asks that vertical-scrolling content work at a width equivalent to **320 CSS pixels** without two-dimensional scrolling (www.w3.org). The WCAG target-size criterion sets a **24 by 24 CSS pixel** minimum for pointer targets (www.w3.org), and platform guidance sets larger touch-target sizes, for example 48x48 dp on Android (support.google.com).

Frameworks and tools agree on structure more than on numbers: breakpoints are ranges, not device widths. Bootstrap describes its breakpoints as a subset of common sizes (getbootstrap.com), Wix defines them as screen size ranges (support.wix.com), and Figma's help page treats a breakpoint as the width below which the

next layout applies (help.figma.com). The best source for which widths matter to a given business is its own analytics. The worksheet in this report maps a client's top five viewport widths to the nearest frame. All sources were retrieved on 2026-11-18.

Introduction and Background

Anyone commissioning a marketing site or app mockup eventually has to answer a deceptively simple question from the designer: "What widths do you want these drawn at?" Searches for this topic return many posts that repeat 375, 768 and 1440 with no stated origin. This report does something narrower and more checkable. It lists each number a reader is likely to meet, names who published it, says what it is for, and labels whether it is a default, a recommendation, a requirement or a measurement. Every figure was read on the publisher's own page on **2026-11-18**.

What a frame width is, and what it is not

A mockup frame is a rectangle drawn in a design tool at a chosen width, in units that stand for CSS pixels. The CSS pixel is a reference unit that MDN ties to 1/96th of a CSS inch (developer.mozilla.org), and browsers map several hardware pixels to each CSS pixel on dense screens (developer.mozilla.org). So "375" in a design file means 375 CSS pixels, not the 750 or 1125 hardware pixels a modern phone may have.

Three widths are easy to confuse:

- **Screen width:** MDN documents `screen.width` as the screen's width in CSS pixels, and notes that not all of it may be available to the window (developer.mozilla.org).
- **Viewport width:** MDN defines the viewport as the area of the window in which web content can be seen (developer.mozilla.org). Its `innerWidth` property includes the vertical scroll bar when one is present (developer.mozilla.org).
- **Window width:** on desktops, tablets with split-screen, and resizable app windows, the available width is set by the window, and Android states that physical devices do not guarantee a specific window size class (developer.android.com).

A CSS media query evaluates the viewport width, and the W3C Media Queries Level 4 specification describes breakpoints as defined on the viewport width using min and max queries (www.w3.org). A frame in a design file is therefore a sample of a continuous range, not a device.

Four kinds of numbers

The sheets below are grouped by what the number is for:

- **Sheet A, framework defaults:** CSS breakpoints shipped by Tailwind CSS and Bootstrap.
- **Sheet B, design-tool and site-builder presets:** widths in Figma Sites, Webflow, Framer and Wix.
- **Sheet C, platform window classes:** Android's width classes, and Apple's size-class approach, which carries no pixel values.
- **Sheet D, analytics shares:** StatCounter's October 2026 resolution charts.

The retrieval date for each is 2026-11-18.

Sheet A: Framework Defaults

CSS frameworks publish breakpoints so that developers share a vocabulary. They are defaults that can be changed, and they are written in terms of minimum widths. A mockup width taken from a framework is a convenience, not a standard.

Tailwind CSS

Tailwind's documentation states that there are five breakpoints by default, "inspired by common device resolutions" (tailwindcss.com). The values are **sm 640 px** (tailwindcss.com), **md 768 px** (tailwindcss.com), **lg 1024 px** (tailwindcss.com), **xl 1280 px** (tailwindcss.com) and **2xl 1536 px** (tailwindcss.com). Tailwind calls its system mobile-first (tailwindcss.com), which means unprefixed styles cover the smallest screens and prefixed styles take over at a breakpoint and above. Its documentation advises using unprefixed utilities to target mobile and overriding them at larger breakpoints (tailwindcss.com). A default can also be removed by resetting its value (tailwindcss.com).

The mobile-first caveat matters for briefs. In Tailwind, "sm" does not mean "small screens"; it means "at 640 px and wider". A developer using Tailwind therefore has no named breakpoint for a 390 px phone: that width is simply the base layout.

Bootstrap 5.3

Bootstrap's documentation describes breakpoints as customizable widths that determine how the layout behaves across device or viewport sizes (getbootstrap.com). The tiers are: below 576 px for portrait phones (getbootstrap.com), **576 px** and up for landscape phones (getbootstrap.com), **768 px** and up for tablets (getbootstrap.com), **992 px** and up for desktops (getbootstrap.com), **1200 px** and up for large desktops (getbootstrap.com), and **1400 px** and up for larger desktops (getbootstrap.com).

Bootstrap labels these tiers with device names, but its own text qualifies them: the breakpoints represent a subset of common device sizes and viewport dimensions (getbootstrap.com). They are mostly based on minimum viewport widths (getbootstrap.com), and the stated goal is "Mobile first, responsive design" (getbootstrap.com). They can be customized through Sass (getbootstrap.com). Material UI follows the same logic: each breakpoint value is a minimum width at which the breakpoint starts, and the defaults can be changed (mui.com).

What Sheet A tells a brief writer

Tailwind and Bootstrap disagree on every threshold except 768 px. A reader should treat a framework breakpoint as the width where the developer's layout rules change, not as a width a designer must draw. Draw frames at widths where the design is meant to look right, and let the developer choose where the code switches. MDN makes the same point: designers now need to build sites defensively instead of catering for a few screen sizes (developer.mozilla.org).

Sheet B: Design-Tool and Site-Builder Presets

Design tools and site builders ship presets so that a new frame starts at a sensible width. These are product defaults that can be edited, and the vendors' pages say so.

Figma Sites

The Figma Sites help article states that a user can choose a desktop, tablet or mobile breakpoint with predefined widths, or enter a custom width (help.figma.com). The preset values **Desktop 1280, Tablet 800, Mobile 375** appear on that page only in the alt text of an image showing the breakpoint dropdown (help.figma.com). They are therefore not stated in the article's running text, which is why that citation carries no text fragment. The same article's worked example uses a different set: a desktop breakpoint 1280 px wide, a tablet breakpoint 800 px wide and a mobile breakpoint **340 px** wide (help.figma.com). It explains the semantics: a screen over 1280 px uses the desktop layout (help.figma.com), a screen between 800 px and 1280 px uses the tablet layout (help.figma.com), and a screen under 800 px uses the mobile layout (help.figma.com).

Three points follow. The 340 in the example is an illustration, not a preset. A breakpoint is a range, so a 1280 frame covers widths above it, not only 1280. And widths are editable after creation (help.figma.com). Another Figma Sites article notes the desktop breakpoint is the primary one by default but can be changed (help.figma.com), and Figma's course page says a site can have as many breakpoints as it likes, but most sites need no more than a desktop, tablet and mobile set (help.figma.com).

For Figma Design (the general drawing tool, as distinct from Sites), the official help article says frame presets exist for popular device and asset templates (help.figma.com). That page lists no pixel widths, and no official Figma page opened for this report states them. A reader who wants the Figma Design preset widths should read them from the Frame tool's sidebar and write the numbers into the brief.

Webflow

Webflow's breakpoints documentation is explicit about its semantics. A site starts in the desktop breakpoint (help.webflow.com). The tablet breakpoint applies to screens **991 px** wide and below (help.webflow.com), mobile landscape to **767 px** and below (help.webflow.com), and mobile portrait to **479 px** and below (help.webflow.com). Webflow also offers three optional larger breakpoints at 1280, 1440 and 1920 px (help.webflow.com); the 1440 px one applies to screens 1440 px wide and above (help.webflow.com). Webflow recommends starting on the desktop breakpoint and moving down the device spectrum (help.webflow.com).

Webflow's figures are max-width rules going down from desktop, while Tailwind and Bootstrap use min-width rules going up from mobile. The same visitor width can therefore land in different tiers in different systems.

Framer

Framer's academy lesson says the width shown on the canvas is the minimum viewport width at which that breakpoint becomes active (www.framer.com). The page gives no default widths that could be quoted. It advises adding a breakpoint when the layout starts to fail, rather than by habit (www.framer.com), and notes that some pages need only desktop and phone while denser layouts may benefit from a tablet state

(www.framer.com). Framer's blog observes that its marketplace templates already shift at widths like 320, 768, 1024, 1200 and 1920 px (www.framer.com). That is an observation about templates, not a product default. Content-driven breakpoints can land on odd values such as 642 px or 911 px (www.framer.com).

Wix Studio

Wix's help page for the Studio editor (the page states it refers to the new Studio Two Editor released in November 2026) (support.wix.com) defines breakpoints as screen size ranges that represent the possible sizes of visitors' devices (support.wix.com). The default mobile breakpoint covers 320 to 1023 px (support.wix.com), an optional tablet breakpoint covers 701 to 1023 px (support.wix.com), and an optional wide-screen breakpoint covers 1920 px and above (support.wix.com).

What Sheet B tells a brief writer

Nothing in these pages recommends a width as the correct one. They offer starting points and say, in different words, to change them when the design needs it.

Sheet C: Platform Window Classes

Platforms publish width classes to help app developers decide when to change layouts. They are useful context for a mockup of a web app, but they describe windows, not devices.

Android

Android describes window size classes as a set of opinionated viewport breakpoints for designing, developing and testing responsive layouts (developer.android.com). The width classes switch at **600 dp** (developer.android.com), **840 dp** (developer.android.com), **1200 dp** (developer.android.com) and **1600 dp** (developer.android.com). The page's table labels the large class as large tablet displays (developer.android.com) and the extra-large class as desktop displays (developer.android.com).

The unit is the density-independent pixel, which Android describes as a virtual pixel unit roughly equal to one pixel on a medium-density screen (developer.android.com). The Android numbers are in dp, not CSS pixels, and they apply to app windows.

The key caveat is stated in the documentation: window size classes are explicitly not determined by the size of the device screen (developer.android.com), but by the window size available to the application (developer.android.com). A foldable, a tablet in split-screen, and a resizable desktop window can all change the class. Android's testing advice is to check all window sizes, especially at the compact, medium and expanded breakpoint widths (developer.android.com).

Apple

Apple's Human Interface Guidelines take a similar approach without publishing point widths on the Layout page opened for this report. They say apps need to adapt to different display sizes, orientation changes, window sizes and multitasking states (developer.apple.com), and to determine layout based on size classes, not device type or orientation (developer.apple.com). Size classes describe the actual space available, whether the app is in portrait or landscape (developer.apple.com).

The Apple page gives no iPhone width table, so it cannot be used as the source of "390" or any other device width. Among the sources opened for this report, 390 appears only as a StatCounter mobile resolution (390x844, 5.33%) (gs.statcounter.com).

What Sheet C tells a brief writer

Platform classes confirm the general pattern: layout decisions follow available space, not device models. If the deliverable is a web mockup, the Android thresholds are one reference for tablet and large-window behaviour, and Apple offers no numbers. If the deliverable is a native app, the platform's own tools are the right starting point, as the last section explains.

Sheet D: Analytics Shares

Analytics shares measure what visitors used, in aggregate, across a sample. They are the only measurements in this report, and the only numbers that can change month to month.

What StatCounter publishes

StatCounter's desktop chart covers October 2025 to October 2026 (gs.statcounter.com), its mobile chart is labelled October 2026 (gs.statcounter.com), and its tablet chart is labelled October 2026 (gs.statcounter.com). The top entries are collected below.

Table 1 below lists the leading entries from the three charts, with the platform shares from StatCounter's separate device-type page.

Platform	Top resolutions (October 2026, share of page views)	Platform share, October 2026
Desktop	1920x1080 27.82% (gs.statcounter.com) 1536x864 9.13% (gs.statcounter.com) 800x600 8.85% (gs.statcounter.com) 1366x768 7.48% (gs.statcounter.com)	38.51% (gs.statcounter.com)
Mobile	414x896 12.67% (gs.statcounter.com) 360x800 8.57% (gs.statcounter.com) 384x832 7.24% (gs.statcounter.com) 390x844 5.33% (gs.statcounter.com)	59.9% (gs.statcounter.com)
Tablet	1280x800 7.96% (gs.statcounter.com) 800x1280 7.89% (gs.statcounter.com) 768x1024 7.39% (gs.statcounter.com)	1.59% (gs.statcounter.com)

Two things stand out in Table 1. First, the mobile and tablet charts are fragmented: the leading mobile resolution has a 12.67% share, and the fourth has 5.33%. No single phone width dominates, so no single phone frame can be "the" mobile width. Second, the desktop chart contains an entry at 800x600 with 8.85%, ahead of 1366x768, a size that is unusual for a current desktop. StatCounter's pages do not explain it, and this report draws no conclusion about its cause. The practical reading is that a single chart entry should not drive a design decision without a check against the client's own data.

Why these shares are not a design target

- **Screen versus viewport is not stated.** The FAQ says StatCounter analyses the browser, operating system and screen resolution for each page view (gs.statcounter.com). The word "viewport" does not appear on the chart, FAQ or About pages that were read for this report, so whether a figure is the physical screen, a scaled value or the browser viewport is not documented there. Any claim that these are viewport widths would be an inference.
- **Sampling.** StatCounter says its tracking code is installed on more than 1 million sites (gs.statcounter.com). It bases its statistics on page views, not unique visitors (gs.statcounter.com), so a heavy-traffic site counts for more than a small one.
- **Bot handling and regional samples.** StatCounter says it identifies bots and prevents them from being recorded (gs.statcounter.com), and cautions that some small regional samples may not be statistically robust (gs.statcounter.com).
- **Revisions.** After a 45-day period from first publication, no changes are made to the data (gs.statcounter.com). October 2026 data read on 2026-11-18 is therefore still inside the revision window.
- **Device classes.** A 1920x1080 desktop entry says nothing about how wide the browser window was.

Taken together, StatCounter data can show that desktop screens cluster around a handful of resolutions and that phones are fragmented. It cannot say which frame width a particular marketing site should be approved at.

Data Analysis and Evidence

Table 2 below is the master sheet: each width, who published it, what it is for, the type of statement it is, and the retrieval date. Rows are ordered by width. "Default" means a product or framework setting that can be changed, "recommendation" means published guidance, "requirement" means a conformance criterion, and "measurement" means an observed share.

Width	Publisher	What it is	Type	Retrieved
320 CSS px	W3C, WCAG 2.2 SC 1.4.10 (www.w3.org)	Width at which vertical-scrolling content must reflow without two-dimensional scrolling	Requirement (test width, not a frame preset)	2026-11-18
320 to 1023 px	Wix Studio (support.wix.com)	Default mobile breakpoint range	Default	2026-11-18
340	Figma Sites help (help.figma.com)	Mobile breakpoint in a worked example	Illustration	2026-11-18
375	Figma Sites help (image alt text) (help.figma.com)	Mobile preset in the breakpoint dropdown	Default	2026-11-18
479 and below	Webflow (help.webflow.com)	Mobile portrait breakpoint	Default	2026-11-18
576	Bootstrap 5.3 (getbootstrap.com)	First media-query tier (landscape phones)	Default	2026-11-18

Width	Publisher	What it is	Type	Retrieved
600 dp	Android (developer.android.com)	Start of the medium window width class	Recommendation	2026-11-18
640	Tailwind CSS (tailwindcss.com)	sm breakpoint (min-width)	Default	2026-11-18
767 and below	Webflow (help.webflow.com)	Mobile landscape breakpoint	Default	2026-11-18
768	Tailwind CSS (tailwindcss.com) Bootstrap 5.3 (getbootstrap.com)	md breakpoint (min-width)	Default	2026-11-18
800	Figma Sites help (image alt text) (help.figma.com)	Tablet preset	Default	2026-11-18
840 dp	Android (developer.android.com)	Start of the expanded window width class	Recommendation	2026-11-18
991 and below	Webflow (help.webflow.com)	Tablet breakpoint	Default	2026-11-18
992	Bootstrap 5.3 (getbootstrap.com)	lg breakpoint (desktops)	Default	2026-11-18
1024	Tailwind CSS (tailwindcss.com)	lg breakpoint (min-width)	Default	2026-11-18
1200	Bootstrap 5.3 (getbootstrap.com)	xl breakpoint	Default	2026-11-18
1200 dp	Android (developer.android.com)	Start of the large window width class	Recommendation	2026-11-18
1280	Figma Sites help (image alt text) (help.figma.com) Tailwind CSS (tailwindcss.com) Webflow (optional) (help.webflow.com)	Desktop preset; xl breakpoint; opt-in larger breakpoint	Default	2026-11-18
1400	Bootstrap 5.3 (getbootstrap.com)	xxl breakpoint	Default	2026-11-18
1440 and above	Webflow (help.webflow.com)	Optional larger breakpoint	Default	2026-11-18
1536	Tailwind CSS (tailwindcss.com)	2xl breakpoint	Default	2026-11-18
1600 dp	Android (developer.android.com)	Start of the extra-large window width class	Recommendation	2026-11-18

Width	Publisher	What it is	Type	Retrieved
1920 and above	Wix Studio (support.wix.com)	Optional wide-screen breakpoint	Default	2026-11-18
1920x1080	StatCounter (gs.statcounter.com)	Top desktop resolution, October 2026 (27.82%)	Measurement	2026-11-18
414x896	StatCounter (gs.statcounter.com)	Top mobile resolution, October 2026 (12.67%)	Measurement	2026-11-18

Table 2 makes three patterns visible. First, the same width appears under different labels: 768 is "md" in Tailwind and "tablets" in Bootstrap, while Figma Sites uses 800 for its tablet preset. Second, the semantics differ even when the numbers look alike. Tailwind's 1280 starts a range, Figma Sites' 1280 is the desktop breakpoint that anything wider also uses, and Webflow's 1280 is an opt-in layer above the desktop base. Third, only one row type is a measurement, and its numbers are screen-style resolutions of a specific sample, not frame recommendations.

Where the numbers cluster

There is partial convergence, which is useful for choosing a short list. Phone widths in the 320 to 414 range appear in Framer's blog (320 to 360 for most portrait phones) (www.framer.com), in Figma Sites (375, with 340 in an example) (help.figma.com) and in the StatCounter mobile chart (360, 384, 390, 393 and 414). Wix's tablet range runs from 701 to 1023 (support.wix.com). Within that range, the frame widths that recur in Sheets A and B are 768 (Tailwind and Bootstrap) and 800 (Figma Sites). Desktop tiers begin around 1024 to 1280 in every framework.

A short, defensible frame set is therefore one phone width in the 360 to 414 band, one tablet width of 768 or 800, and one or two desktop widths between 1280 and 1920. This is this report's own reading of the sheets, not a claim made by any single publisher, and the choice within each band is a project decision.

How to Choose Widths for Your Audience

Use the client's own analytics first

Aggregate shares describe the world. A brief should be written against the audience of the specific site. The method is to pull the top viewport widths from the site's analytics for the last 90 days or a full seasonal cycle, split by device category, and map each to the nearest frame. Because analytics tools differ on whether they report screen size or viewport size (StatCounter's pages, for example, do not say which they report), the first step is to read the tool's definition. Where it reports physical screen sizes, divide by the device pixel ratio to approximate CSS pixels, and treat the result as an estimate. If the site is new and has no analytics, use the closest comparable site, and say so in the brief.

A worksheet

Table 3 below shows the worksheet structure with hypothetical numbers. The mapping rule is this report's proposal, not a published standard: assign each viewport width to the largest frame that is not wider than it, and flag any width that falls more than about 60 px from its frame for an in-between check.

Rank	Top viewport width (Hypothetical Example)	Share of sessions (Hypothetical Example)	Nearest frame	Gap to frame	Action
1	1440	31%	1440	0	Approval width
2	390	24%	390	0	Approve mobile
3	1366	12%	1280	86	Check in between
4	412	9%	390	22	Covered by 390
5	768	6%	768	0	Draw tablet

In this hypothetical case, the 1366 width is 86 px above the nearest frame, so it needs an explicit in-between check rather than its own frame. The numbers are invented for illustration; a real client replaces all of them with their own data.

Mapping audience type to a starting set

- **B2B and desktop-heavy audiences:** analytics often show desktop dominating weekday sessions, so the approval width is usually a desktop frame, with tablet and phone drawn as derived layouts. Webflow's own advice is to start designing on the desktop breakpoint and move down (help.webflow.com).
- **Consumer and mobile-heavy audiences:** StatCounter's worldwide platform chart shows mobile at 59.9% for October 2026 (gs.statcounter.com), which is a worldwide aggregate and not a statement about any one business. Where a client's own data shows the same tilt, approve the phone frame first.
- **Mixed audiences:** approve two frames, one phone and one desktop, and treat tablet as a verified in-between width.

The Brief Clause and Acceptance Test

Suggested clause

The sources above support one claim: widths are chosen by the project, so the brief must choose them. A clause that does the job:

> "Design frames will be delivered at 1440 px (desktop, approval width), 768 px (tablet) and 390 px (phone) wide. The approved layout is the 1440 px frame; the phone frame is approved separately. The finished page must not break at any width between 320 px and 1920 px, including widths between the delivered frames."

The three frame widths and the approval width are placeholders drawn from the clustering analysis, to be replaced with the client's own worksheet results. The final sentence turns "in between widths must not break" from a hope into a testable statement.

Why include 320

WCAG 2.2 Success Criterion 1.4.10 (Reflow, Level AA) requires that vertical-scrolling content can be presented without loss of information or functionality, and without two-dimensional scrolling, at a width equivalent to 320 CSS pixels (www.w3.org). For horizontally scrolling content the equivalent test is a height of 256 CSS pixels (www.w3.org). W3C's note ties 320 CSS pixels to a 1280 CSS pixel window at 400% zoom (www.w3.org), so the test serves low-vision users who zoom a desktop browser as well as small-phone users.

A 320 px frame does not need to be drawn. It needs to be tested, and the clause names it.

Acceptance test, without a pass/fail gate

This section covers only how to word the width criterion in an approval. It does not replace a wider build-readiness review. An approver can run these width-specific checks on a built page or a prototype:

- **Intermediate widths:** drag the browser window from 1920 px down to 320 px slowly. Layout changes should happen at a width, not leave a band of widths where elements overlap or scroll sideways. Breakpoints can land on odd values, so test between frames.
- **Text wrapping:** headlines and buttons should wrap without clipping. WCAG's Resize Text guidance expects content to remain usable at up to 200 percent enlargement (www.w3.org), and says that enlargement must still be possible where a media-query breakpoint changes text size (www.w3.org).
- **Tap-target size:** WCAG 2.2 sets a minimum of 24 by 24 CSS pixels (www.w3.org), and the enhanced criterion uses the larger 44 by 44 CSS pixel measure (www.w3.org). Apple's accessibility page lists a 44x44 pt default control size (developer.apple.com). Google's Android accessibility guidance recommends touch targets of at least 48x48 dp (support.google.com).
- **Zoom behaviour:** at 400% zoom on a 1280 px window, the page should reflow to the 320 CSS pixel layout with no two-dimensional scrolling (www.w3.org).
- **Real devices:** check at least one physical phone, because MDN documents that mobile browsers may render at a virtual viewport of 980 px without the viewport meta tag (developer.mozilla.org).

Who draws and who tests

The designer draws the frames named in the brief, and the client signs off the approval width. The developer tests in-between widths. Both statements belong in the brief, because a mockup approved at one width says little about a build tested at five. Where the work is bought as a service, for example through a flat-rate design

and development subscription, the same clause applies unchanged: the widths are the client's decision, and the service executes them. For a single one-off page, a freelancer or in-house designer is often the simpler route.

When the Need Is Different

The widths in this report are for web pages and web apps. Other deliverables follow other specifications.

- **Native app design:** use the platform's own size-class guidance. Android documents width classes at 600, 840, 1200 and 1600 dp (developer.android.com), and Apple's guidance is to determine layout from size classes (developer.apple.com).
- **Email:** Mailchimp's help says to set email width to 600 px or less (mailchimp.com). Campaign Monitor says a good maximum width to aim for is 600 pixels (www.campaignmonitor.com). This is email-client practice, not a web breakpoint.
- **Display ads:** the IAB's New Ad Portfolio calls itself an industry recommendation (www.iab.com) and lists fixed sizes such as 728x90 as "In Transition" to aspect-ratio units (www.iab.com). Google Ads Help names 1200 x 628 as the most common image size and says requirements vary by campaign type (support.google.com). Check each publisher's current specs.
- **Social share images:** Facebook's documentation recommends images of at least 1200 x 630 pixels for high-resolution devices (developers.facebook.com).

None of these is a screen width. Keep them out of the frame list in a website brief and list them as separate asset specifications.

Implications and Future Directions

Window-based thinking is spreading. Apple's guidance frames layout as available space across display sizes, window sizes and multitasking states (developer.apple.com), and Android's classes follow the window, not the device (developer.android.com). If the web follows, "device width" will matter less and "available width" more, which is the position MDN and the framework documentation already take. A brief that names a handful of frames, an approval width and a no-breakage range will age better than one that lists device models.

Two practical implications follow for people commissioning work:

- **Write the numbers into the brief.** A brief that says "standard breakpoints" invites a different reading from each tool.
- **Date the evidence.** StatCounter's chart for October 2026 will be revised for 45 days and replaced by a new month. Any shares quoted in a brief should carry the month and the retrieval date.

Limits of this report are worth stating. Material Design's own pages could not be read in the retrieval environment, so Android's documentation is the only Material-derived source used. Figma Design's frame-preset widths and Framer's default widths were not found on official pages, and are therefore not listed. No

search-volume or demand data was available for this topic, and the clustering analysis above is a reading of published numbers, not a statistical result. The Figma Sites preset values appear only in image alt text on its help page, so they are the least firmly sourced numbers here.

Frequently Asked Questions (FAQs)

What screen sizes should I design for?

Choose three or four frame widths yourself, and name one as the approval width. The sources opened for this report publish defaults and measurements, not a standard. Tailwind says its defaults are "inspired by common device resolutions" (tailwindcss.com), and Bootstrap says its breakpoints are a subset of common sizes (getbootstrap.com).

What Figma frame sizes should I use for web design?

For Figma Sites, the preset widths are Desktop 1280, Tablet 800 and Mobile 375 (help.figma.com), and they can be edited (help.figma.com). For Figma Design, the help article lists frame preset categories without widths (help.figma.com), so read the numbers in the tool and write them into the brief.

What are the common responsive breakpoints?

Tailwind uses 640, 768, 1024, 1280 and 1536 (tailwindcss.com). Bootstrap 5.3 uses 576, 768, 992, 1200 and 1400 (getbootstrap.com). Webflow uses 991, 767 and 479 going down, with optional 1280, 1440 and 1920 (help.webflow.com). Android's width classes switch at 600, 840, 1200 and 1600 dp (developer.android.com).

What desktop mockup width should I use?

A frame between 1280 and 1440 is a reasonable starting point, with a check at 1920; this is a recommendation of this report, not a published standard. StatCounter's October 2026 desktop chart lists 1920x1080 as the largest share at 27.82% (gs.statcounter.com), but its pages do not say whether that is screen or viewport width (gs.statcounter.com).

Is the mobile mockup width 375 or 390?

Either works if the brief names it. 375 is the Figma Sites mobile preset (help.figma.com) and a width marker Framer mentions for iPhones (www.framer.com), while 390x844 holds 5.33% of StatCounter's October 2026 mobile chart, behind 414x896 and 360x800 (gs.statcounter.com). No source opened for this report publishes 390 as a preset. Test at 320 as well (www.w3.org).

What tablet frame size should I use?

768 or 800. Figma Sites presets 800 (help.figma.com), while Tailwind and Bootstrap start their tablet-like tier at 768 (tailwindcss.com). Tablets were 1.59% of StatCounter's October 2026 platform share (see Table 1), so a tablet frame may be optional for sites whose own analytics show little tablet traffic. A worldwide aggregate is not a statement about any single business, so check the client's analytics before dropping it, and test the layout at tablet widths either way.

What are the most common screen resolutions?

On StatCounter's October 2026 desktop chart, 1920x1080, 1536x864, 800x600 and 1366x768 lead, and on mobile 414x896 and 360x800 lead (see Table 1). These are shares of page views and are subject to a 45-day revision window (gs.statcounter.com).

How many breakpoints should a website design have?

Three is a common default: Figma says most sites need no more than a desktop, tablet and mobile breakpoint (help.figma.com). Add a fourth only when the layout fails at a width the first three do not cover (www.framer.com).

Conclusion

No standard frame width exists. The numbers in circulation come from four different kinds of publisher doing four different jobs: CSS frameworks (Tailwind at 640 to 1536, Bootstrap at 576 to 1400), design tools (Figma Sites at 1280, 800 and 375, Webflow at 991, 767 and 479 with optional larger layers), platforms (Android's 600 to 1600 dp window classes, with Apple offering size classes and no pixel table), and analytics (StatCounter's October 2026 shares). Each is labelled in the master sheet by type and retrieval date, and only the last is a measurement.

For a brief, the useful move is to stop asking for a standard and to state a decision. List three or four widths, one of which is the approval width, and say that widths in between must not break. Anchor the in-between statement to the 320 CSS pixel reflow test (www.w3.org) and the tap-target sizes published by W3C and Google (www.w3.org) (support.google.com). Choose the specific frames by mapping the client's own top viewport widths to the nearest frame, using the worksheet, and re-read any third-party share on the day the brief is written, because the figures are revised for 45 days.

Source links

Links cited in this article, in order of first appearance. Full addresses are provided for readers using extracted text.

1. https://tailwindcss.com/docs/responsive-design
2. https://getbootstrap.com/docs/5.3/layout/breakpoints
3. https://help.figma.com/hc/en-us/articles/31242797809815
4. https://developer.android.com/develop/ui/compose/layouts/adaptive/use-window-size-classes
5. https://gs.statcounter.com/screen-resolution-stats/desktop/worldwide
6. https://gs.statcounter.com/screen-resolution-stats/mobile/worldwide
7. https://gs.statcounter.com/faq
8. https://www.w3.org/TR/WCAG22
9. https://www.w3.org/WAI/WCAG22/Understanding/target-size-minimum.html
10. https://support.google.com/accessibility/android/answer/7101858
11. https://support.wix.com/en/article/studio-two-editor-working-with-breakpoints
12. https://developer.mozilla.org/en-US/docs/Glossary/CSS_pixel

13. https://developer.mozilla.org/en-US/docs/Web/HTML/Guides/Viewport_meta_element
14. https://developer.mozilla.org/en-US/docs/Web/API/Screen/width
15. https://developer.mozilla.org/en-US/docs/Web/API/Window/innerWidth
16. https://www.w3.org/TR/mediaqueries-4
17. https://mui.com/material-ui/customization/breakpoints
18. https://developer.mozilla.org/en-US/docs/Learn_web_development/Core/CSS_layout/Responsive_Design
19. https://help.figma.com/hc/en-us/articles/31242788601879
20. https://help.figma.com/hc/en-us/articles/35895705647127
21. https://help.figma.com/hc/en-us/articles/360041539473
22. https://help.webflow.com/hc/en-us/articles/33961260724243
23. https://www.framer.com/academy/lessons/breakpoints
24. https://www.framer.com/blog/responsive-breakpoints
25. https://developer.android.com/training/multiscreen/screendensities
26. https://developer.apple.com/design/human-interface-guidelines/layout
27. https://gs.statcounter.com/screen-resolution-stats/tablet/worldwide
28. https://gs.statcounter.com/platform-market-share/desktop-mobile-tablet
29. https://www.w3.org/WAI/WCAG22/Understanding/reflow.html
30. https://www.w3.org/WAI/WCAG22/Understanding/resize-text.html
31. https://developer.apple.com/design/human-interface-guidelines/accessibility
32. https://mailchimp.com/help/about-html-email
33. https://www.campaignmonitor.com/dev-resources/guides/design
34. https://www.iab.com/wp-content/uploads/2019/07/IABNewAdPortfolio_FINAL_0719.pdf
35. https://support.google.com/google-ads/answer/13676244
36. https://developers.facebook.com/docs/sharing/webmasters/images
37. https://gs.statcounter.com/about

THE PUBLISHER

About Tapflare

Tapflare gives companies an on-demand design and web team for a flat monthly fee.

How it works

Each subscriber has a dedicated project manager who learns the brand, assigns every request to a senior specialist in that kind of work, and reviews the result before it is delivered. Requests are submitted and discussed in the Tapflare portal, each plan includes a daily allotment of design or development time, and the client keeps full ownership of the export and source files.

What it covers

Graphic design, custom illustration, presentation design, logos and brand guides, video editing and motion graphics, website and mobile-app mockups, and website development, from design mockups to HTML, to drag-and-drop builders or custom code.

What it is not

Tapflare is not a marketing agency, a media-buying, SEO or copywriting service, a software product or a freelance marketplace. Where a reader needs one of those, the honest answer is to say so.

Visit

See Tapflare and its pricing.

Articles published here explain general practice and cite their sources. Prices and plan contents, including Tapflare's own, change; check the provider's current pricing page before deciding.

Website: https://tapflare.com

This article is educational. Verify consequential medical, legal, financial, regulatory and technical decisions with appropriate professionals and the cited primary sources. Company services are described separately from the article's research findings.