



Michał Malewicz & Diana Malewicz

Designing User Interfaces

Special Edition of the free chapters 🕶️

Get the full book at:

www.designingui.com

Table of Contents

Contents of the full book. Checkmarks next to chapters in this version.

INTRO

1. Introduction	3	4. Design and Art	21
2. Who is this book for?	7	5. What makes UI good?	26
3. UI vs UX	14	6. Perception	34

DESIGN BASICS

7. Screens <input checked="" type="checkbox"/>	47	16. Tables	213
8. Layout and grid	55	17. Forms	229
9. Objects <input checked="" type="checkbox"/>	77	18. Modals, popups	261
10. Colors	94	19. Navigation	280
11. Gradients	125	20. Animation	299
12. Typography	139	21. Photos	311
13. Icons	167	22. Illustration	323
14. Buttons <input checked="" type="checkbox"/>	178	23. Language	329
15. Cards	203		

BONUS CONTENT

24. Design styles	337	28. Developer handoff	394
25. Design process	351	29. Prototyping	404
26. Design systems	358	30. Summary	415
27. UI Audit	373		

WHAT YOU GET IN THE FULL BOOK

These 50+ pages are an excerpt of the full Designing User Interfaces Ebook. The main difference between it, and other books is that it's a "living ebook".

We are going to add new content (both chapters, video and other materials) every month until the end of 2020. And if you have the full book you will get those extra materials for FREE!

The full book contains:

- ✓ 420 pages of knowledge, tips, tricks and techniques
- ✓ UI Design Styles mini-ebook for FREE (\$30 value)
- ✓ .Sketch, .Figma and .XD files with icon packs and more.
- ✓ Exclusive video materials.

Buy the full book now:

<https://gum.co/WsCMA>

OUR OTHER PRODUCTS



UI DESIGN STYLES - \$30

UI Design styles is a special ebook covering all the modern trends in design with examples and how-to's on how to achieve these specific trends. It covers everything from Soft UI, through Material Design, Neumorphism, Dark Mode and more.

If you own *Designing User Interfaces* you get this book for FREE.



FRONTEND UNICORN - FREE CHAPTERS

Good design is one thing, but a bad developer can ruin even the best projects. This is why we're creating this special book for Front-End Developers that will help them get to the next level, earn more and do a way better job!

You can sign up for discounts and free downloads at:

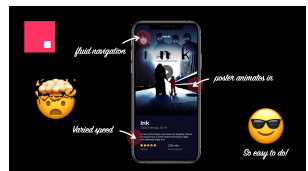
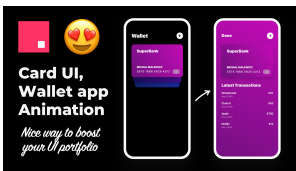
<https://frontendunicorn.com>

Free tutorials on YouTube

<https://youtube.com/c/MalewiczHype>

Subscribe to our YouTube channel to get access to even more FREE content including:

- ✓ UI Animation Tutorials
- ✓ Design Tips
- ✓ Design Audits
- ✓ Live redesigns
- ✓ The basics of UX design



1.

Free

Chapters

Screens

THE HISTORY

The UI design is all about displays. Based on the fact that the UI is everything we see, it's essential to understand what do we see it on. In the early days of the web, it was quite simple - you designed somewhere between 640 x 480 and 800 x 600 pixels. We can, of course, go back even farther - to the first Macintosh or Xerox's very first Desktop-UI but those UIs, but let's assume old school color CRT's as a base here.

But that was the late 90's, and things started rapidly changing after that. Most of the displays of that era had similar resolution ranges, were super heavy, and couldn't display too many colors. They also had low refresh rates that resulted in subpar scrolling experience. The displays of that era often limited what websites could achieve visually.



Before today's flat panels, most of the UI's were displayed on old-school CRT monitors. They often had a bit of a blur to the image (even at higher resolutions), but in most cases, you could easily distinguish individual pixels.

“

**Always design for
the base screen
resolution or @1x.”**

THE BASICS

The technology leaped forward since the early internet days. Nowadays, the phone you are holding in your hand has a resolution that didn't exist twenty years ago. They also display millions of colors and have super high refresh rates.



A quick comparison. A typical late 90's CRT display had about 1/6 the resolution of the current iPhone that's much smaller in size.

PPI - PIXELS PER INCH

At some point, 72 pixels per inch became the display standard. That all changed with the introduction of the iPhone 4 with its retina display. The premise was the pixels were so dense you couldn't easily see the individual ones anymore. A base (or 1x) resolution was the same as the previous iPhone, but the pixel density was a multiplier of that resolution resulting in sharper text and images.

The iPhone 3GS had a 320 x 480 base resolution, which became the 1x size. The iPhone 4 doubled the number of pixels to 640 x 960, but the actual design of elements was kept at 320 x 480 and upscaled on the device.

After the iPhone 4's 2x pixel count, we started seeing 3x, 4x, and larger pixel densities in both phones, tablets, and laptops.

Since the UI's we design are all inside vector-based tools, we don't need to worry about actual resolutions anymore. The base resolution of 1x serves as a template that works on low-density screens and is sharper and more precise in high-density ones.

For example, the modern iPhone X used a 375 by 812 base resolution, but its actual pixel count is three times larger at 1125 by 2436.

To give an example - if you're designing a button, you need to make it at least 44p high. That means that designing at 1x it will be 44 pixels (at 1x a pixel is the same as a point) while at 2x it will be 88 pixels, but still 44 points in your design.



Most popular 1x screen sizes in 2019 are:

1440 x 900
1280 x 800

Typical laptops

375 x 812

Most popular
iPhone

360 x 640

Most popular
Android Phone

1024 x 768

Most popular
Tablet

BUT YOU DON'T HAVE TO REMEMBER THIS.

Most design tools have already created presets while starting a new design that caters to all the latest devices you might need. Those resolutions are then set automatically at 1x in every case.

FRAGMENTATION

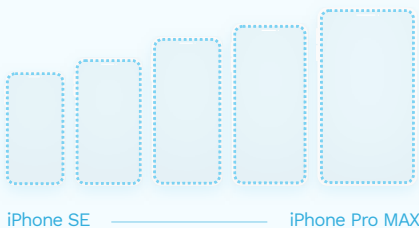
Sadly a growing number of screen resolutions end up with a very fragmented display landscape. We design for TVs, laptops, tablets, phones, watches, and IoT devices, which requires a lot of planning and device-specific modifications for the design to work.

The first questions to ask when starting a design is what kind of screen it is going to work on and what's the typical viewing distance.

A TV app should have higher contrast and more significant UI elements than a mobile phone app, mostly because it's often used from across the room, while a phone app is just inches away from the face.

DOZENS OF PHONE SCREENS

But the place where fragmentation hurts the most is from within one device category. The most ubiquitous one - the mobile phone - has so many potential resolutions and aspect ratios that there is no way to make one design fit all. There are dozens of screen resolutions outlined in the Google Play store for Android-based mobile devices. Apple used to have a more consistent set a few years ago, but has since then abandoned the path and went towards individual

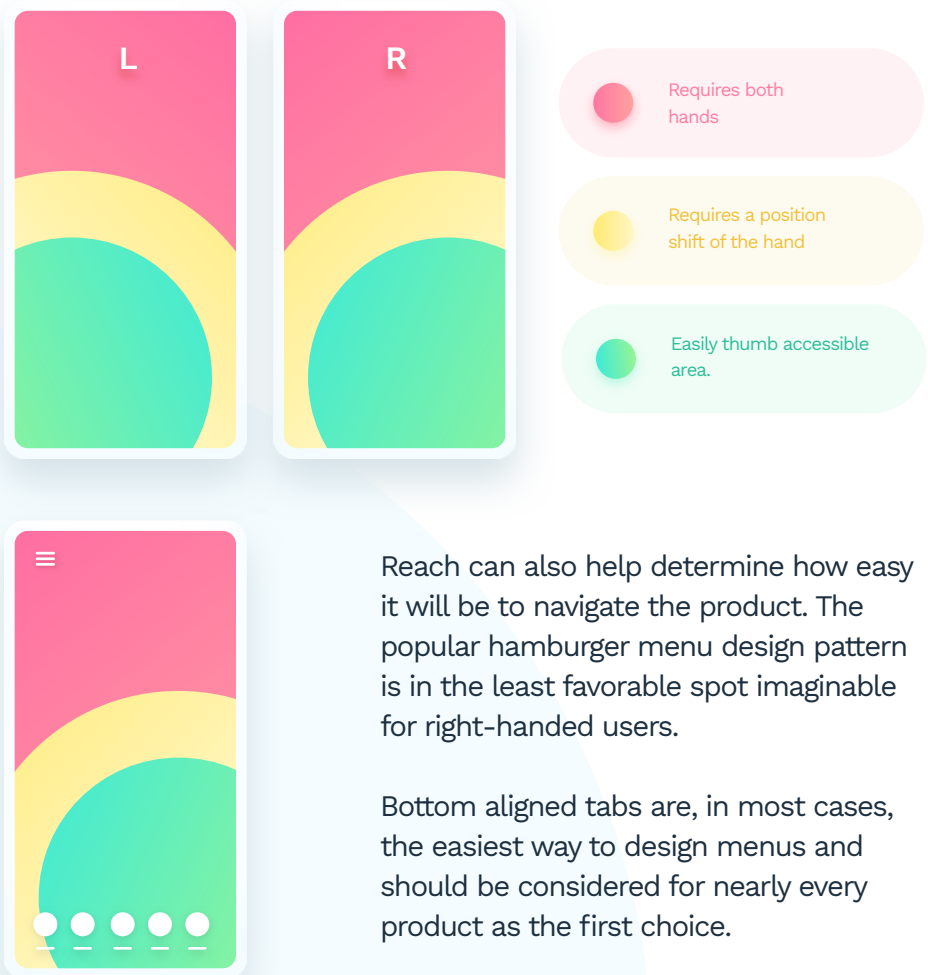


There are five current generation screen resolutions for just the iPhone and countless more for Android phones.

RANGE AND REACH

In mobile devices, it is also essential to think about the average reach. A randomly pieced together interface can be complicated to use with one hand and lead to frustration.

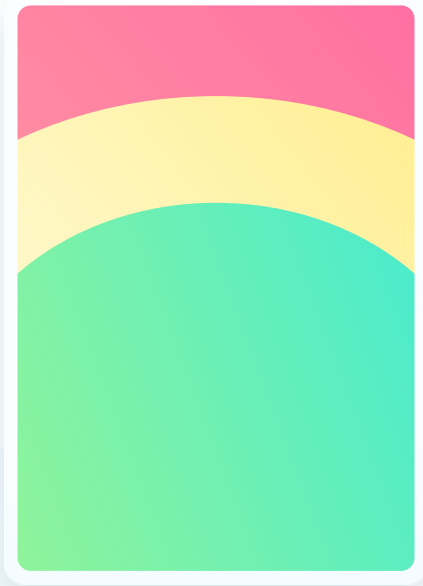
We assume that the typical phone usage pattern is a single hand holding the phone with the thumb of the same hand doing most of the on-screen work.



Reach can also help determine how easy it will be to navigate the product. The popular hamburger menu design pattern is in the least favorable spot imaginable for right-handed users.

Bottom aligned tabs are, in most cases, the easiest way to design menus and should be considered for nearly every product as the first choice.

We usually use our tablets differently. The most common way is one hand holding the device, while the other one is actively interacting with the UI. It dramatically increases the range both in portrait and landscape orientations.



The most common usage pattern for tablets is using both hands. It allows for a different approach to the interface planning that merges the larger reach area with the fact of having a much larger canvas.

In the rare cases of holding a tablet with both hands, the average thumb reach extends on both sides of the device. This pattern works great for split keyboards and gaming controls.

Objects

THE BASICS

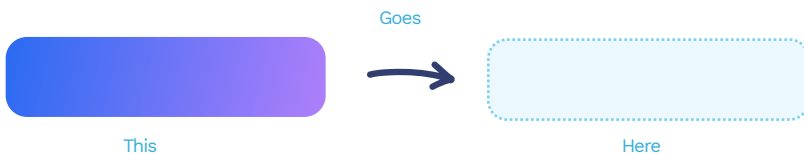
All currently available UI design tools work by manipulating vector shapes. That means the forms themselves are represented by numbers that define them and can be stretched and enlarged without losing quality.

Their parameters are just numbers that generate a new shape according to the criteria with each change.



If you're planning to design interfaces, be prepared that most of the time, it will be about moving various rectangular objects around. Rectangles are the most popular shape (right before ellipses) in all the interfaces.

The general idea of UI design is about moving rectangles around in just the right way.



This chapter covers the very basics of all the UI design ingredients.

“

**UI Design is
moving rectangles
around. ”**

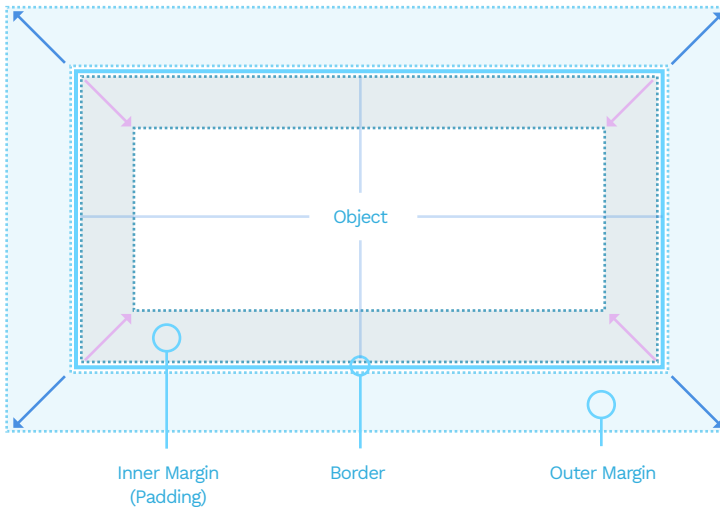
THE BOX MODEL

The Box model is the most basic way of defining a digital interface object both in design and in code. Most of the design we do works according to the box model, so it's essential to get to know it well. It consists of four main elements:

- ✓ Fill
- ✓ Border
- ✓ Outer margin
- ✓ Inner margin

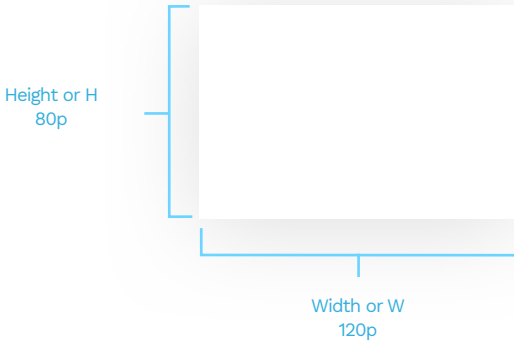
The inner margin is also called padding. It means the more extensive it is, the bigger the safe area inside the object becomes. In the example below, it's the grey area inside the rectangle.

Outer margin (often just called "margin") is the area outside our object. It allows us to position it with enough safe space concerning everything around it. It's the blue area outside the rectangle below.

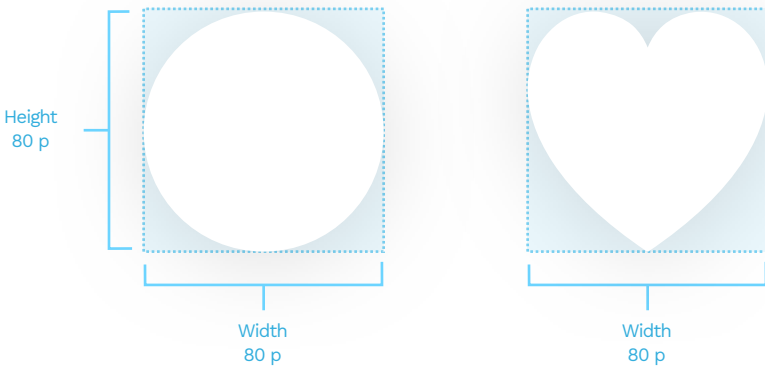


SIZE

Width and height (in points) define the size of an object. In most cases, we use W and H for short. As it's a two-dimensional space, the width is a scale on the horizontal axis, while the height is a scale on a vertical one.

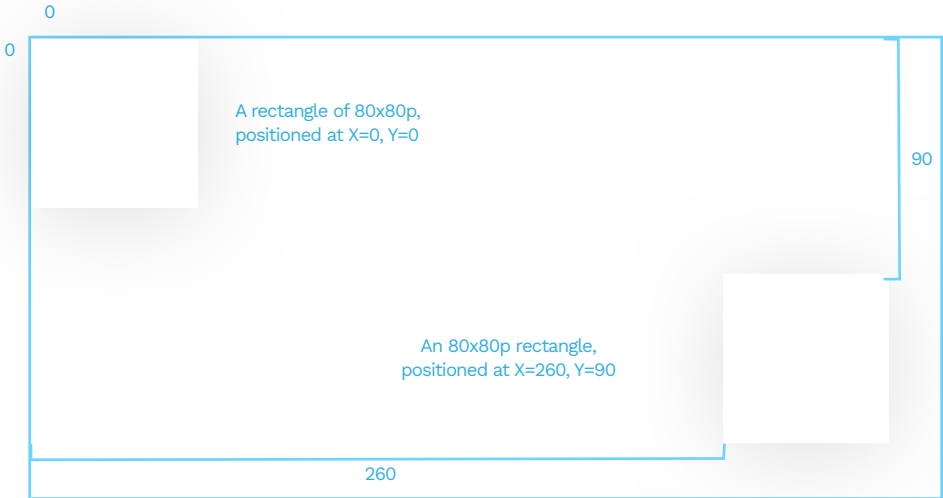


The width and height are always of the smallest possible rectangular box that can contain our object - even if the object itself has an irregular shape.

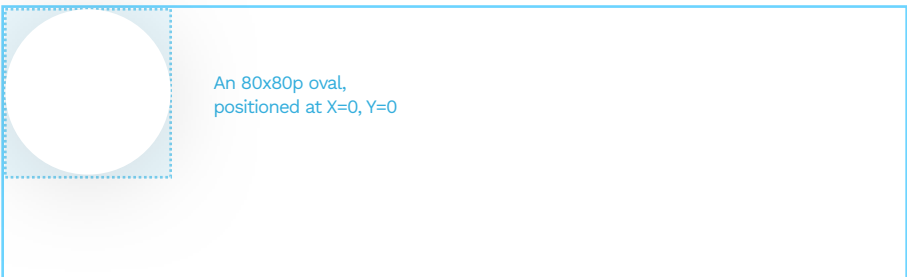


POSITION

The position of the object is a set of numbers on the X and Y axis, defined by the artboard that contains it. The X is the position on the horizontal axis, while the Y is the position on the vertical one.

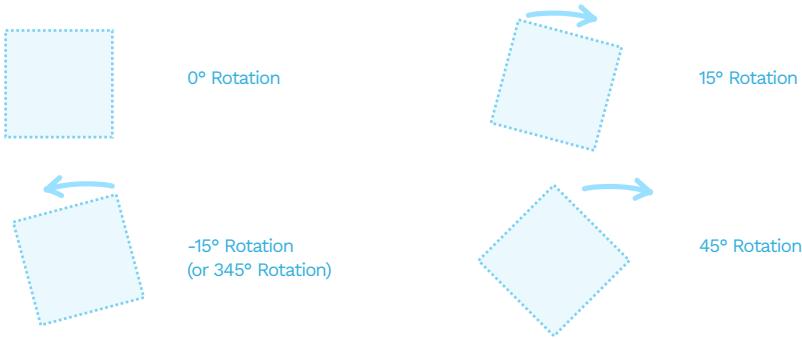


Irregular objects positions are the position of their bounding box that contains the actual shape within.



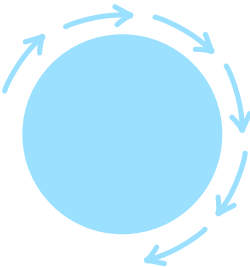
ANGLE

Angle defines the rotation of our object going clockwise - the default being 0° . Rotation can also be a negative number. It's worth remembering that a -15° rotation is the same as $360^\circ - 15^\circ$, which equals 345° .



If we want to have a consistent interface, it's best not to rotate objects by hand, but instead enter the right angle from the keyboard as it is way more precise than manually turning the object.

If we Flatten the object, it resets the angle to 0, but it is still rotated - only the new perspective becomes the base for further rotation.



A perfect circle will look the same no matter of its angle.

BORDER-RADIUS

A border-radius is a number. Just like the width and height, it's also presented in points. The larger that number, the larger the roundness of the corners of our shape. You can use it for all of the edges at once or set them individually.

In general, even slightly rounded corners (between 2 and 6p) are considered more friendly than sharp ones (of 0 p). However, it's good to have consistency in the roundness. The grid base unit that we select should define the border-radius. It should also match its surroundings.



Starting from the left a border-radius of : 0p , 2p , 6p , 100p

LEARN MORE



A square with only two corners rounded.



A square with each corner having a different border-radius.

Different border radii in the same object can make a great button if it also matches the brand guidelines (i.e., the logo also has a similar pattern).

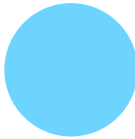
If you decide to make your buttons that way, remember to be consistent with them across all the screens.

FILL

Fill is another name for the object's background. It can be a color, a gradient, or a photo. Each of these can also have a different level of opacity. If an object doesn't have a fill, is also lacking a border or an effect, it won't be visible in the interface as it requires some defining characteristics to be visible. You'll still be able to select it, however, move it around, and modify it. It will also still exist in the layers list.



No Fill



Color Fill



Gradient Fill



Photo Fill

You can combine more than one fill on the same object - for example, adding a transparent color to a photo will colorize the result so it can match the rest of your UI.



Photo



Photo + Color



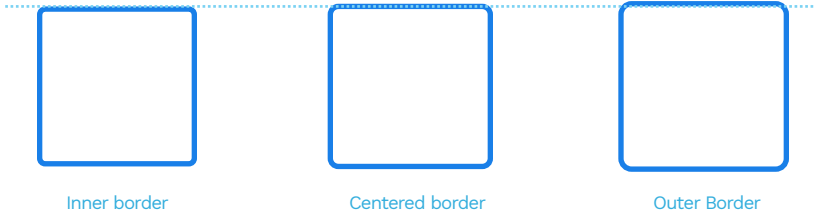
Photo + Gradient



Photo with a
vanishing gradient.

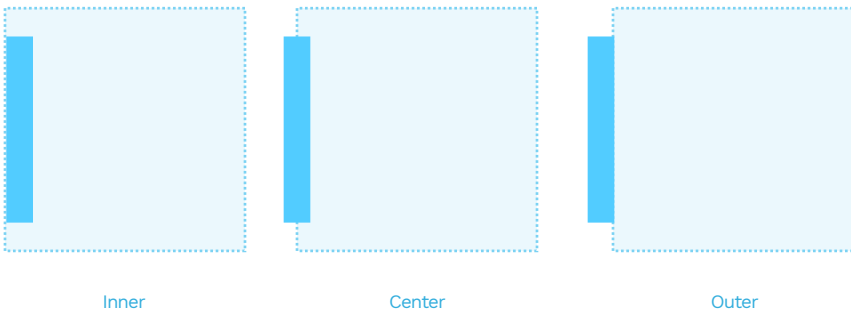
BORDER

The border is a line that goes around our object. It can go inside (inner border), outside (outer border), or between them (center). Remember that only the inner border doesn't visually make the object larger.



Above, you can see a 60p sized square with a 2p border around it.

From the left: Inner-border, centered, and outer. As you can see, only the first one is still that 60p size visually.



A 2p border width that's centered adds 1 point inside and one outside of our element. That makes our square 62x62. The outer border adds 2 points to each side, which makes the square 64x64.

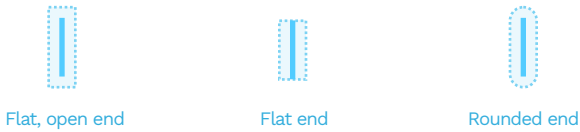
BORDER

The border can have different weights (width) in points and can be a dashed or dotted line. We can also fill it with both a color and a gradient.



Most of the design tools allow you also to modify the endings and joints of your lines.

The ends can be open, flat, or rounded while the joints can be sharp, rounded, or angled.



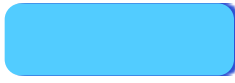
DROP SHADOW

Outer shadows (or drop-shadows) are easily the most common effect used in UI. A typical shadow relies on an offset from center (x, y, or both) a blur and an opacity value.

Some tools also allow for a "spread" value, which makes the shadow look like a smaller element is casting it.



X=0, Y=2,
Blur= 0, Spread= 0

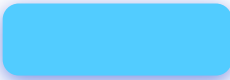


X=2, Y=0,
Blur= 0, Spread= 0

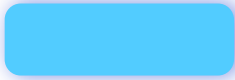


X=2, Y=2,
Blur= 0, Spread= 0

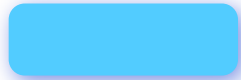
Blurring our shadow makes it more natural. It's essential, however, that the opacity value is never higher than 40%.



X=0, Y=2,
Blur= 8, Spread= 0

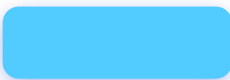


X=2, Y=0,
Blur= 8, Spread= 0

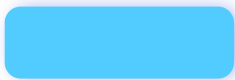


X=2, Y=2,
Blur= 8, Spread= 0

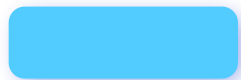
A negative spread value makes our shadow look natural.



X=0, Y=2,
Blur= 8, Spread= -2



X=2, Y=0,
Blur= 8, Spread= -2



X=2, Y=2,
Blur= 8, Spread= -2

“

Using a drop shadow helps with establishing a hierarchy between our interface layers.”

DROP SHADOW

A natural-looking shadow is one of those elements that make the most impact in design. The most important part of looking natural is avoiding pure black shadows and using a shadow derived from our primary color instead. Pure black makes the contrast ratio too big to look natural. If you study real-life shadows, you'll notice they often vary in shade and tone.



Our base color



Shadow color
Saturation -10
Brightness -20



Base color and
shadow combined.

If our primary color is green, then a green button can case a blurred dark-green shadow. With a big enough blur and a negative spread value, we end up with a good looking result.



Too sharp, dark shadow creates too much contrast. That high contrast black is often a default value for shadows that we should always modify.



If we change the black color to a darker shade of our primary color, it starts to look more natural and softer already.



With a negative spread value, we eliminate the shadow spillover, making it look more realistic.

INNER SHADOW

Inner shadows are relatively rare in UI. It has the same parameters as a drop shadow, but it appears inside of the object.



Outer shadow



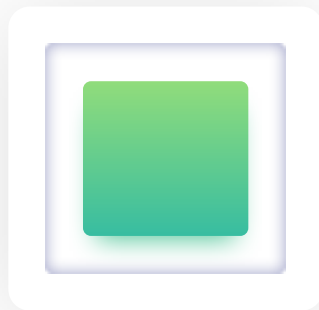
Inner shadow

They are not as popular because most interfaces are a series of layers stacked on top of one another. In that case, an outer shadow makes sense as it provides the depth. An inner shadow would suggest the object has a hole in it.

The only use-cases of this style are form inputs (both form fields and checkboxes or radio-buttons) and extruded shapes in the Neumorphism method. They can be used to make the objects appear more life-like in some instances but should only be used moderately.



Modern UI's are using the shadows to build a visual hierarchy and layered structure of elements.



An inner shadow makes us think the object has a hole in it, which breaks the expected layering structure and can be confusing.

GAUSSIAN BLUR

Most design tools nowadays have a Gaussian type blur that extends the effect in every direction evenly. Its primary value is the radius. The larger it gets, the more prominent the blur effect.

Gaussian blur is the most often used UI effect. You can employ it into transitions between screens, or show a bit of realistic depth of field by selectively blurring the background.



0



2p



8p



16p



This type of blur can also help you generate non-standard, point-shadows under objects. The easiest way to achieve it is by blurring an ellipse and placing it under the object casting the shadow. You can either use it on its own or combine it with a standard drop shadow for an even more unique result.

BACKGROUND BLUR

Background blur became popular when Apple started using it in their OS to achieve that behind smoked-glass effect in some screens. An object with this effect applied blurs everything under it.



MOTION BLUR

A motion blur simulates the movement of an object in a direction defined by the angle value.



No blur



0° Angle



45° Angle



90° Angle

ZOOM BLUR

A zoom blur happens when the object becomes blurry from the inside out. It's often used in photography, but not a great pick for interface design.



Blur value 0



Blur value 25



Blur value 50

In this particular blur type, you can also set the origin of the blur. By moving that point around, you can achieve some interesting effects.



Origin in the top
left corner



Origin in the center



Origin outside
of object bounds

Buttons

THE BASICS

A button is an interactive element that results in an action described in it. You can bet that if it says "save" on a button, clicking it will most likely "save" something. It's also an essential interactive element of a digital product. It's crucial to get buttons right.

It can lead to a purchase, downloading, sending, and many other vital actions. Digital buttons are also descendants from real-world buttons like in the tv remote, record player, or game controller.



BUTTONS SHOULD LOOK LIKE BUTTONS

The most important rule while designing a button is for it to stand out enough so it won't be confused with anything else.



By removing elements from a button, it's function starts to dissolve and disappear. It becomes decoration or text, losing its actionable qualities.

“

**A button should
look like a button.”**

FAMILIAR SHAPES

We are used to specific shapes and forms that are typically associated with an action. The more our button looks similar to what we associate with buttons - the better. It is the main reason why a rectangle (or a rounded rectangle) is always the safest choice for a button.



These objects will be identified as a button right away.

An exception to the rule is a text link. A link has the same general function as a button, only lacking the shape. Due to its online popularity, it is also really straightforward for the users. Blue and purple text links - dating back to the beginning of the world wide web are one of the most common UI patterns imaginable.

DOWNLOAD

DOWNLOAD

Blue and purple text links are one of the fundamental design truths. Most users understand them right away.

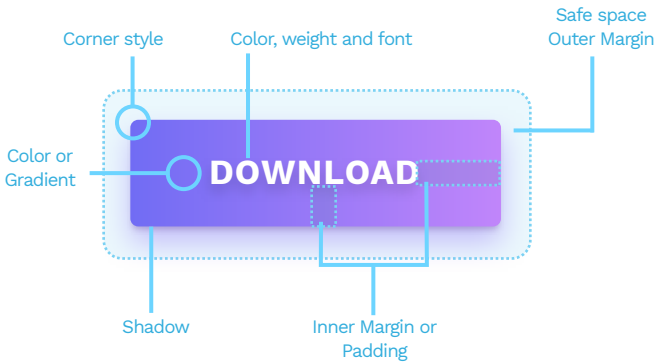
Other shapes and forms (triangle, circle, organic) are not as recognizable to the user. Please proceed with caution and use them only when the general style of your product requires deviating from the norms.



Elements that can take much longer to understand they are actionable.

BUILDING A BUTTON

While designing buttons, remember about all of these elements and choose them wisely. Using the brand book as a baseline, think about what kind of buttons will match the brand and fit well within the interface.



you should set the padding and the safe space using your grid base numbers. (see the layout and grid chapter and "the red square method")

in the above example, the left inner margin is twice the size of the vertical one, which is a safe choice for increased readability.



ALIGNMENT

Poorly aligned and unevenly spaced buttons are one of the most common problems of all interfaces. Double-check if your button labels are centered both horizontally and vertically. Create guides if you need to be sure.



Aside from grid-based methods, there's a safe way of choosing button spacing with multiples of capital W. If at least 1 W fits on each side of the button label, you're safe.

On the sides, it's even better to use 2 x W for increased readability. In general, the smaller the padding inside the button, the harder it is to read and understand.



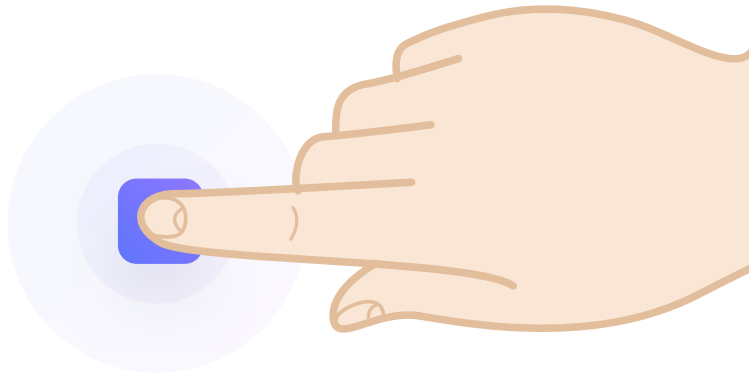
Don't forget about the space around your buttons. If you have a group of them, the safe area should be individual for each one - try not to overlap it!



SIZE MATTERS

Both web and mobile buttons should also have a set minimum size. If your buttons are too small, it will be difficult to tap or click on them. That results in frustration and can lead to users uninstalling your app. The best way is to start with 44 by 44 points for all interactive elements on mobile devices. The sweet spot is somewhere around the 50p height for mobile buttons. The width (including the Double W minimum spacing) can be any number equal to or larger than the height.

In the case of cursor-based devices, 32 by 32 should also work. Remember that even on desktops, the larger the button, the easier it is to use.



This button is too small. The inability to quickly tap or click on it will frustrate your users.



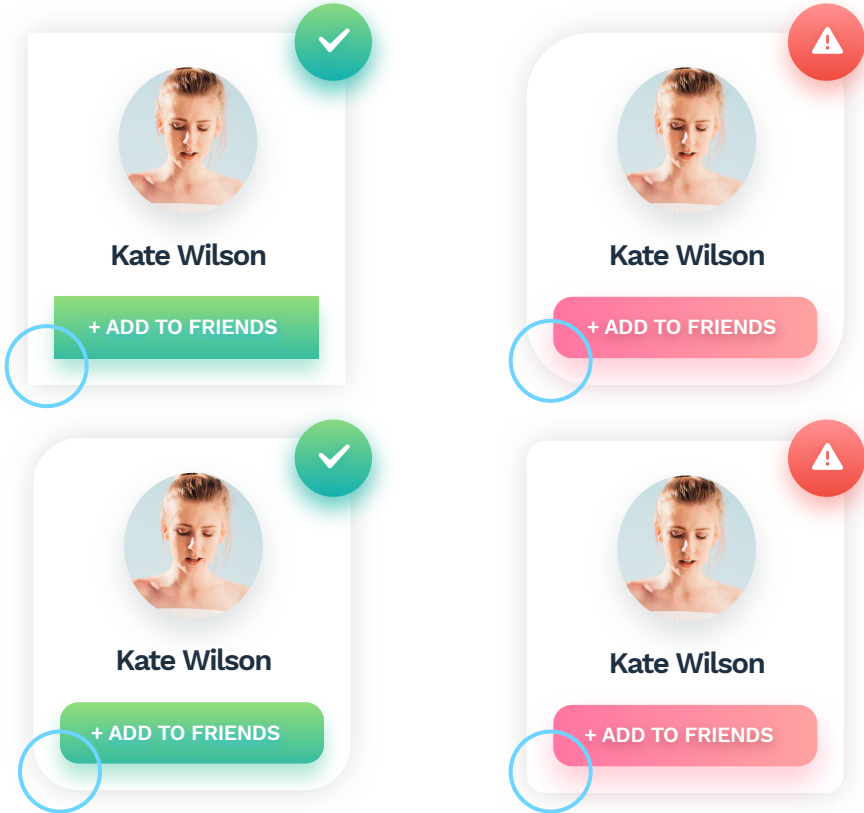
This button is still a bit too small for mobile devices. It probably will be fine on laptops.



This button is big enough for both computers and mobile devices. People should be able to find and activate it with ease.

FIGURE TO BACKGROUND

If you're using rounded buttons, keep in mind to have the right rounded corner ratios towards other on-screen elements. Using the same one for everything will create imbalances in the margins.



The diagonal spacing is the same here like the one on the left side and the bottom. That results in a better looking, cleaner, and faster to process outer edge.

The diagonal spacing is larger (bottom) and smaller (top) than the side spacing. That makes the edge stand out too much and take the attention away from the button itself.

CALL TO ACTION

CTA or Call to Action is an essential, actionable element of a website or app. It's usually closely tied to the sales process.

While designing your CTA button, remember that it has to stand out. The screen's on should not have any other buttons that compete with it visually or catch attention. The best way to achieve that is to make it the only element with a unique color or style on the screen. Don't forget to think about the contrast, so your button doesn't blend with a background too much.

Size also matters. If it's too small, people may miss it while scanning the page. If it's too big, it can suffer from "banner blindness," which is ignoring visual elements that resemble an advertisement.

As with other buttons, the sweet spot is between 40 and 60 points of height, with proper spacing on both sides.



DOWNLOAD THE BOOK!

THE RIGHT MESSAGE

Remember that your CTA needs to "call to action" through wording as well. It needs to be convincing and inviting from the start. It also has to be precise - things like "Sign up for X" or "Download the book" work best.

Try to avoid labels like **click here**, **next**, or **back**.

“

**The optimal height
of a CTA button
is between 40 and
60 points.”**

TEXT LINKS

Text link

Hover



Active



Disabled

A text link was the first way to have an actionable element on a web page. A blue link, underlined is thus the fastest identifiable element of the page. Nearly everyone associates it with action and tries to click it.

With that in mind, remember not to underline non-active elements (especially if they're blue) as an inability to take action can confuse and frustrate the users!

41 SHADES OF BLUE



When Marissa Mayer worked at Google, she tested over 41 different shades of blue to find the best combination available. The goal was, of course, for people to click more, and one particular shade won the race. You can now see it in Google's search results. 41 Shades of Blue may sound like a cheesy novel title, but it also shows the dedication to research that makes a difference.

In truth, however, if your link has enough contrast and differs from its surroundings (through color and underline), you can be sure it will be understood.

BUTTON TYPES



The primary is the button for the main, positive actions on each screen — things like OK, Save or Download.

Unlike the CTA button, you can use more than instance of this button on a screen, but try not to group it with other primary buttons or the CTA.



Design your Call to Actions to be visually different than all the other buttons. You can use colors or sizes to achieve that. The goal is for the button to be visible right away and to catch the attention. Because of that, you should only use one instance of CTA on one page. If you are designing a long page, you can duplicate the primary CTA near the bottom as a "catch-all" type button.

If you try and use more than one "strong" button, your page will be converting less, as the users, attention divides between many elements.

CTA should also deliver the most important actions of your interface like "Buy now" or "Sign up."

BUTTON TYPES



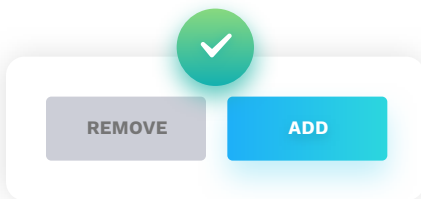
Secondary buttons can be used for all the less critical actions or if we have many buttons with similar importance on one screen. If you're unsure which button to choose, this should always be your safe choice.

Make sure this button does not compete visually with either the primary or the CTA, and you're good to go.

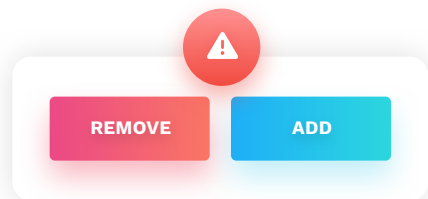


Tertiary buttons are best for negative actions like "cancel" or "revert." You can also use a text link or a ghost button as your tertiary, instead of a rectangle.

There's a notion that all negative actions (like 'remove') should be a red button. In most cases, the tertiary will do the trick. If you're unsure how to design negative actions, think about it like this: Are those negative actions desired? If they're not the core of what your project has to accomplish, then making them red would take attention away from other, more critical on-screen elements.



The negative action is quickly identified as less important than the positive one.



The negative action is more eye-catching than the positive one.

BUTTON TYPES



Icons on buttons are mostly used in apps to condense more functionality within limited space. If there's a text label next to an icon, the perceived importance of that button is higher. In general, however, it's best to avoid having too many buttons with only the icon. Even the most well-known symbols can be interpreted differently by your users and confuse them. One of our most famous examples is the heart symbol. In one particular target group, we researched most of the users thought the heart meant it was a healthy product.

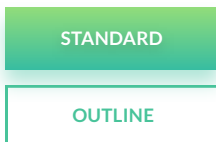


Outline buttons, also called ghost buttons, are usually reserved for less important actions. They're often used as a tertiary button as they don't compete as much visually with primaries and CTA's.

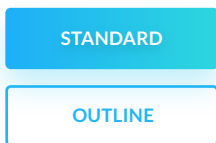
You shouldn't use a ghost button as your Call to Action, because it captures less attention than a button that's filled. Because of that, it takes slightly longer to process, which may lead to lower conversion. Their active states (hover, active) work best when filled with color.

BORDER RADIUS

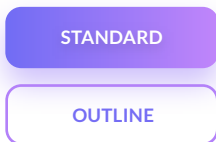
Buttons can also vary through their border-radius. Remember that each project is unique - if your brand image has only sharp corners, having rounded buttons will make the UI inconsistent. Mixing a couple of different border radiuses within one interface can also lead to consistency and readability issues unless done on purpose.



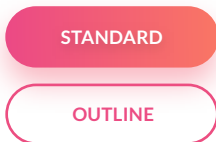
Sharp corners fit brands that have similar shapes in their logo and other elements.



Slightly rounded corners are likely the most popular and also the most user-friendly choice.



Larger border-radius can make it harder to align the text above and below the button.



"Pill" or the highest possible border-radius works well for CTA's.

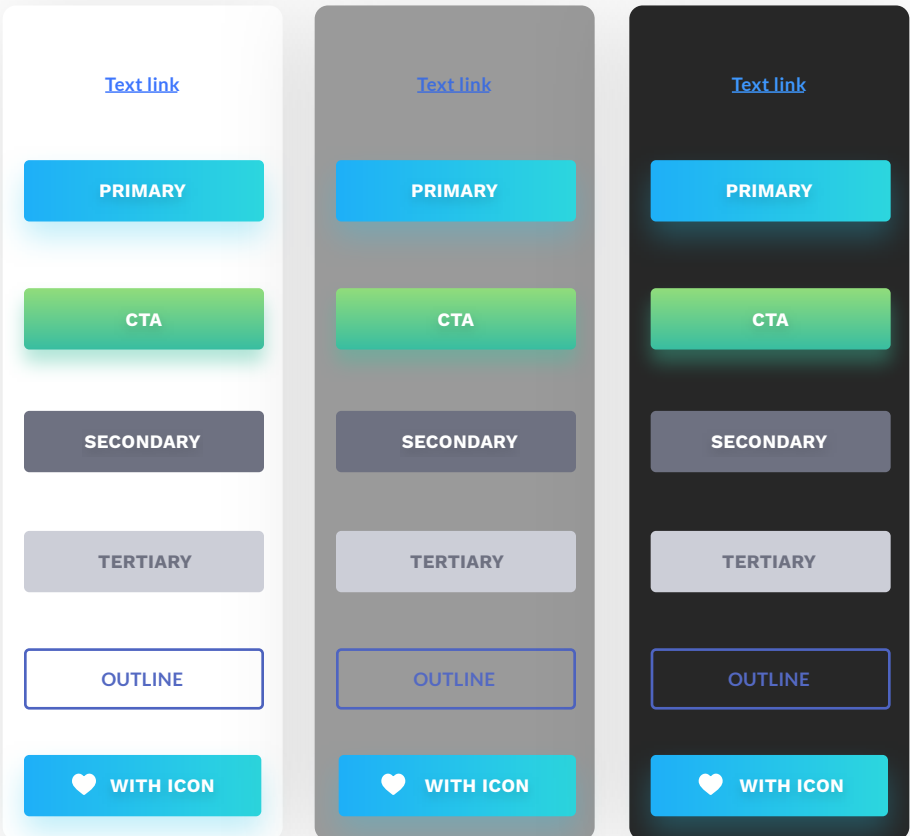
The same rules apply to your icon-buttons.



VISIBILITY

It's good to design your buttons in a way that they work well on various backgrounds. A lot of apps and websites are adding "dark modes" now, which results in at least two completely different versions of the product.

The colors and saturation of the buttons should work well with most of the backgrounds. You can create a simple tester like below and always aim for high contrast and visibility.



BEST PRACTICES

Important buttons also work well with icons. A basket or a cart icon quickly identifies a checkout. However, that is the case only as long as the word checkout is used as a label.



A right-facing arrow or chevron placed after the button label makes the resulting message stronger. The user is more inclined to click and "proceed." It works well if you'd like to make your CTA stronger.



Buttons with shadows are also more "clickable" and noticed much faster than flat ones. Add a subtle drop shadow to make it stand out from the background.



The button on the right is more visible and will be a stronger Call to Action.

BEST PRACTICES

Rounded buttons are considered more friendly and positive than sharp edges. At the same time, however, they make it a lot harder to design content around them. If you have left-aligned text just above the button, the more rounded the corner, the less that text will visually fit. It makes you feel as if the left margin is in two places at the same time.

If the button is only slightly rounded text above, it seems to fit within the perceived left margin.



The text over a more rounded button begins to "bounce" away to the right visually.



Text over a completely rounded button loses visual alignment and seems to be outside the grid.



Pill type buttons work the best when the text above them is centered or removed altogether.

BEST PRACTICES

Aligning icons on buttons is one of the hardest tasks a designer can face. In many cases, it depends on the font-weight to icon weight ratio and their "optical size."

However, you can try and use this simple solution that will work fine in most cases.



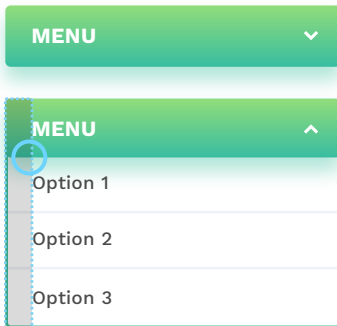
Create a square or a circle the size of our button height. If it's a pill or rather sharp-cornered button, just align it to the right side of the shape. Then create another smaller object inside of it with its spacing being the height of our text. Then fit the icon right in the middle and try not to make it larger than the smaller circle. In the case of chevron, text-height works for chevron height as well.



If our border-radius is larger than 8 points (but not quite a pill), it's best to align our square to the place where the border starts to bend. If you need to use a more complex or larger icon (like a shopping cart), make sure it's not bigger than our smaller container.

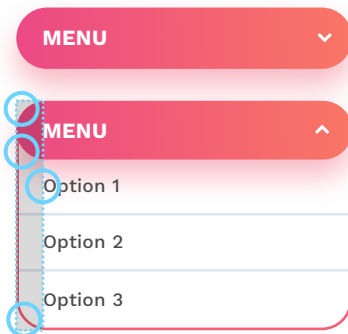
BUTTON STYLE DROPDOWNS

Rounded corners are also problematic while designing a button/dropdown combo. It works the same way as a normal dropdown, but its initial state looks more like a button with a full color or gradient fill. When you expand the button, it creates a shape that distorts the left edge, even more, making it harder to process the text. You can see that clearly in the second image.



If the corners are sharp or only slightly rounded, we end up with a natural, easy to read expanded version.

The text margin allows the user to go down in a straight line without shifting focus away.



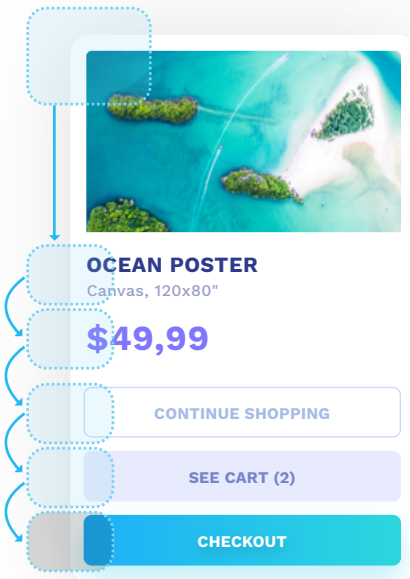
If the corners are completely round, we end up with a shape that captures our attention and swings it between two separate grid lines.

That impairs readability as our focus shifts from edge to edge while scanning the image.

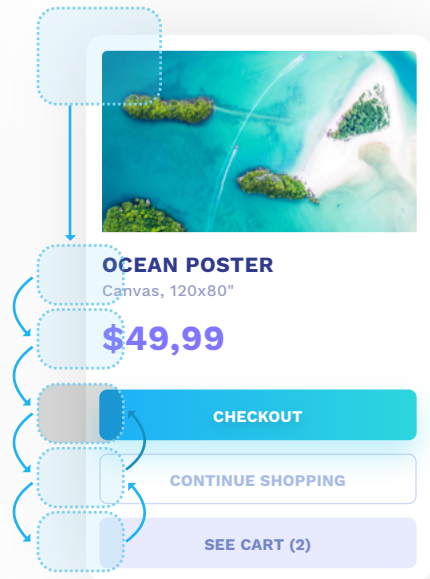
HIERARCHY

Buttons should be in places where the user expects them to be. That usually means below the descriptive elements and often aligned to one of the sides.

That matches the way we scan things - from top to bottom along the left edge (which is also called an F-pattern). If you want to stay safe without bending the rules, pick the first image.



By scanning the card from top to bottom (along the left edge), we naturally end up on a strong element, which in this case, is our CTA button.



If the most pronounced visual element is not at the bottom, our eyes travel back and forth along the edge, which makes for slower processing.

AUTO-ALIGNMENT IN DESIGN TOOLS

If we want to center our label on a button correctly, we need to make sure that both the label and button height are either even or odd numbers. If one value is an odd number and the other one is an even one, you will never be able to center them correctly.

We can try to use auto-alignment tools like auto-center, but sometimes even they end up missing a pixel or two.

Sadly entering an even number for both the container and label height is often not enough. Fonts are interpreting the numbers differently (depending on the font construction) and render the resulting text slightly larger or smaller than we anticipated. That's why try and use little squares to measure the height from the top and bottom (and then from the sides) to be sure it's the same on each side. The buttons below both have an even number as their height and an even number for the font-size. One of them had to be modified by hand to fit perfectly instead of missing one pixel.



We used auto-alignment in both buttons. The example on the right, when checked manually, shows that there's a one-pixel gap on one side. In a case like this, you need to adjust the button height to match the label manually.

“

**Don't trust numbers
alone. Test your main
buttons manually.”**

CONTRAST

The text to background contrast ratio is fundamental to get right. If it's too small, the user can skip it entirely while scanning the screen. If the contrast is too high, it can become an eye-strain - especially with high brightness displays.

Most design tools allow you to add a plugin (often free) to check the contrast ratios between elements. Aim for at least AA, but also use your judgment as to what will work OK. We'll talk more about the contrast in the chapters on colors.

Remember that some of your users may have impaired vision. Even if they don't, why make their lives harder?

Which one do you think will get noticed first?



The one on the left can be barely visible on low contrast screens or by visually impaired users.

TEXT CASE

A label written in all-caps is slightly slower to process than title-case one. It, of course, depends on the number of words, their length, and even individual letters.

Doing buttons ALL-CAPS is not wrong, but try to limit them only to short one or two word labels. Words like "DOWNLOAD" or "SAVE" are safe to use either way. It's also closely tied to how the brand itself communicates in its marketing materials and branding. If it includes ALL-CAPS, it's good to be consistent with those rules.

The title case is slightly more readable, especially with more wordy labels. Try it for things like "Sign Up For Our Newsletter" or "Download the Free Ebook."

Try to avoid entirely lowercase button labels, though, unless they're brand defined.



The label on the left will be slightly faster to read and process. The one on the right will be easier to align correctly inside the button. The choice is yours.



Discount!

**Get a 15% discount when you
sign up for the newsletter at:**

designingUI.com



We build design-driven software that delivers results.

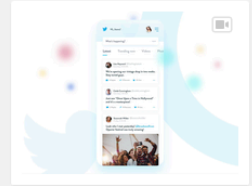
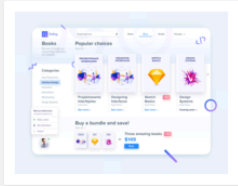
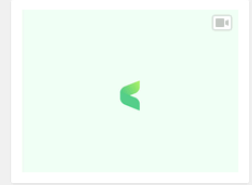
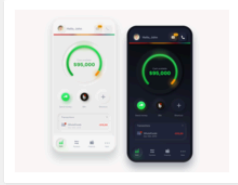
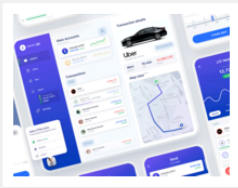
In 2013 we co-founded HYPE4 - a design-driven software house. Since then, we've been working for both small startups and Fortune-500 companies from all over the world. We were lucky enough to be able to work on nearly every product category (from banking, finance, advertising, and media to pregnancy tracking).

We started as a small, design-only studio with three employees. We did the designs, and then external software houses handled the development. We quickly noticed a huge problem with that approach. The execution quality was so low that we were rarely satisfied. In some cases, the designs were butchered beyond recognition. We decided to take action by starting the development branch of the company but strictly following our design-driven approach.

All of our developers understand the design principles, and our designers know a bit how to code. That allows us to achieve a much higher front-end development precision and overall quality. That has since become our defining feature.

At HYPE4, we build web and mobile products of all sizes. After hours we also work on our internal startups and have a lot of fun doing it. If you'd like to learn more, go to:

<https://www.hype4.com>



FOLLOW US ON DRIBBBLE!

Many of the examples and projects made for this book will be published on our Dribbble profile in the coming weeks. Follow to stay in the loop!

dribbble.com/HYPE4