

MVP Design System, Version 0.1

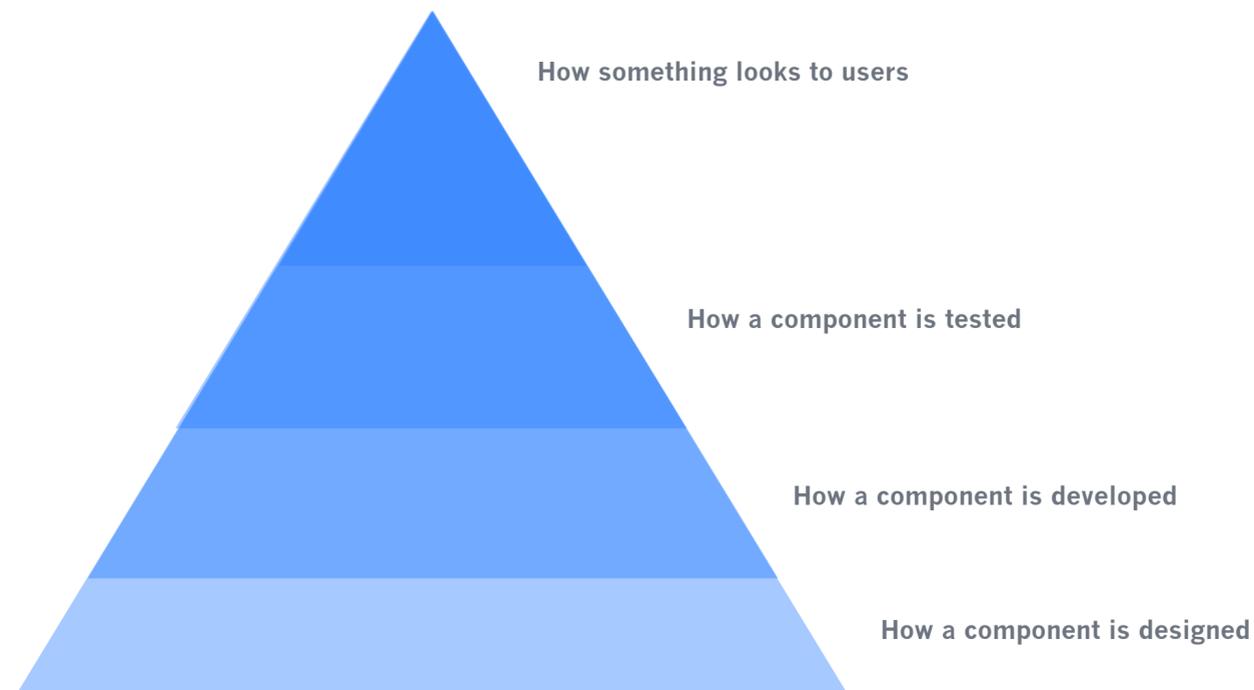
Goals of MVP design system

We want to make designing, testing, and developing easier for everyone, while driving a consistent and modern user experience.

We hope to enable The Co-operators to meet its goals over the next year by facilitating rapid design, building, and testing with a shared design system. We aim to work toward a centralized, living component library with dedicated developers and designers to improve our ability to move quickly, limiting required testing time and allowing us to launch new ideas or make updates fast.

The system aims to make The Co-operators faster and more flexible ahead of brand changes this fall, with shared testing and device standards that drive consistency and speed.

The system should provide a more consistent and trustworthy user experience to users across account management and Quote and Buy, as well as Co-operators.ca.



Who should use this MVP design system?

Designers: Consistency starts with us. We can use this system and ensure our Sketch components match it.

Developers: Developers who are able should use the design tokens as specified exactly in this system, to help lay the foundation for a component library and the style changes coming with new branding this fall.

Testers: Testers should check live pages and work in progress against this system. We're still compiling testing standards, which we hope to standardize to help manage your workload, make it easier to collaborate across teams, and drive consistency. Venkat Reddy on the Co-operators side is working to centralize standards and efforts.

Where will this live? How can I get the latest version?

We hope to publish this to a place easily accessible to vendors, developers, testers, and designers in a live version with code snippets and more detail on digital.cooperators.ca.

We aim to make this a living, breathing system that is frequently updated on digital.cooperators.ca and in Zeplin. Please refer to the latest version of the guide while working; design token names will not change, but their values will be updated with the new brand guidelines in September.

We are forming a design committee consisting of representatives from across Co-op properties and platforms, with developers as a key stakeholder group. We are developing processes to update and deploy this design system so that it saves time and improves consistency.

Table of contents

Best practices and ideas

How should the elements in this design system be used? We can go beyond the colours and fonts listed: we can combine them according to best practices, improving the client

1. Consistency who should use this guide, and what parts?
2. Testing standards (a work in progress)
3. Best practices, design
4. Best practices, copy

The atoms of the design system: tokens

These small building blocks are the foundation of consistency across the design system and using tokens for them makes them easy to update.

5. Why tokens?
6. Fonts
7. Colour tokens
8. Grid and breakpoint information
9. Animation tokens
10. Spacing and inputs

The molecules and organisms of the design system: components

By combining the atoms of the design system, we can make components—modals, cards, buttons, and more.

11. Text fields
12. Checkboxes
13. Radio buttons
14. Progress indicators
15. Buttons
16. Dropdowns
17. Modals

Expected CX & AX design system conformity

Customer Experience Portal

The design team is using the work done on the CX Landing page to inform this Global Design Guide or Design System. They will be developing design component guidelines as more standard components become realized (the landing page is mostly unique components). The Platform Zero team will be working to create a Component Library during or soon after components are built to re-use with subsequent screens. This will both be an ongoing, living set of artifacts.

Advisor Experience Portal

After presentations and discussions with Slalom, we have gotten clarity on how design works within Salesforce. We will be using the Colour tokens from the Global Style Guide once approved.

Other products and platforms

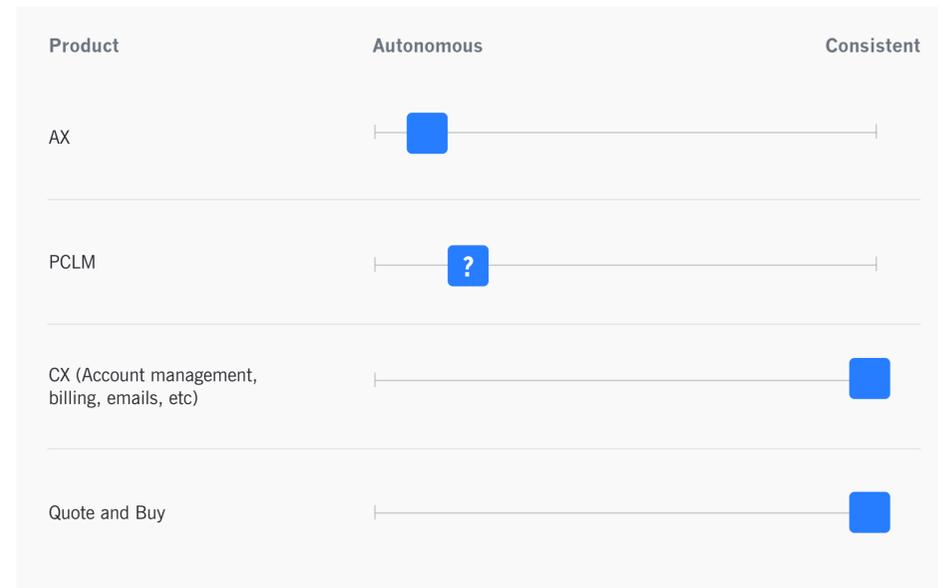
The breadth of sites that fall into this category is large, and for the most part, out of eCX team's control. We hope to drive consistency across all customer-facing products in the future, including commercial, farm, and life.

Projected 2020 use of the design system

Area as defined in this system	Advisor Experience and Business Intelligence	CX (OLS 2.0, Quote and Buy)
Colour tokens	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Type tokens		<input checked="" type="checkbox"/>
Spacing tokens	Set by end user	<input checked="" type="checkbox"/>
Form field interaction patterns (error message, hover, etc)		<input checked="" type="checkbox"/>
Form field look and feel		<input checked="" type="checkbox"/>
Buttons interaction patterns (hover, disabled, etc)		<input checked="" type="checkbox"/>
Buttons look and feel		<input checked="" type="checkbox"/>
Grid		<input checked="" type="checkbox"/>
Accessibility standard		<input checked="" type="checkbox"/>
Testing standard		<input checked="" type="checkbox"/>

Will totally conform

Degree of autonomy in look, feel, and interactions with this design system



Projected future use of design system, in collaboration with coop.ca team (TBD)

Area as defined in this system	AX & BI	Coop.ca	CX, Farm, Commercial	Benefits Now
Colour tokens	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Type tokens		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
Spacing tokens	Set by end user	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
Form field interaction patterns (error message, hover, etc)		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
Form field look and feel		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
Buttons interaction patterns (hover, disabled, etc)		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
Buttons look and feel		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
Grid		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
Accessibility standard		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Testing standard		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	

Testing standards

We want to limit our testing to the devices and browsers our users use the most so we can move faster. Unifying testing standards and centralizing testing efforts across CX will help us share a component library and deliver a consistent experience, because the code will not be affected by two diverging streams of feedback from testing.

Browser testing

Browser name	AX	PCLM	eCX	QB	.ca
Mozilla Firefox, latest stable version minus two versions					
Chrome, latest version minus two versions					
Safari Version 10					
Microsoft Edge 14.19041 minus two versions					
Internet Explorer	We want to choose not to test on this going forward to help us move faster, but this new standard hasn't been confirmed.				

Accessibility standards

Accessibility guide	AX	PCLM	eCX	QB	.ca
WCAG 2.0 AA	 *				
SiteImprove browser extension					
VoiceOver with Safari 10 on macOS 10.12					

In terms of accessibility, we prioritize “function” problems over “friction”, meaning if it’s broken, or users can’t get the information they need or do what they need to do at all, that’s our priority. Ultimately we want the user experience to be as easy as possible, but we need to ensure it’s not broken first, then fix the friction issues after that. - Wendi Phillips, Co-operators Digital Standards

-  Used all the time
-  Sporadically

* Salesforce components are already tested to AA standard. We want to make sure that as we build or diverge, we stick to that standard.

Core device testing (Venkat Reddy is overseeing an effort to centralize and streamline these standards)

Device name	AX	PCLM	CX	.ca
iPhone iOS 11.X (767px and below)				
Android 8.X (767px and below)				
Tablet (768px)				
Desktop Windows 8.1				
Desktop OS X 10.9				

Best practices, design

These are the basic guidelines that designers must consider when working on a user interface.

- **Keep users informed of system status** with constant feedback.
- **Set information in a logical, natural order.**
- **Ensure users can easily undo/redo actions.**
- **Maintain consistent standards** so users know what to do next without having to learn new toolsets.
- **Prevent errors if possible**; wherever you can't do this, warn users before they commit to actions and provide plain-language messaging.
- **Don't make users remember information** – keep options, etc. visible.
- **Make systems flexible** so novices and experts can choose to do more or less on them.
- **Design with aesthetics and minimalism in mind** – don't clutter with unnecessary items.
- **Don't interrupt or give users obstacles** – make obvious pathways which offer an easy ride.
- **Reduce distractions** – let users perform tasks consecutively, not simultaneously.
- **Cluster related objects together.**
- **Avoid jargon.**
- **Don't delay users** – ensure quick interface responses.

To help deliver great interfaces, any feedback from end users is better than none at all. Presenting design drafts and flows to end users for their feedback before the interfaces built, even if it's in an informal survey, is a necessary step to check assumptions and make sure your research has carried through in the design.

Best practices, copy

Resources for writing copy

Our corporate style guide contains details and examples for how to maintain our corporate voice and tone. **A common voice helps assure readers that the messages they receive represent the whole company, regardless of how they receive them.** The tone we use reflects our **friendly, professional expert** advisors and client service staff across the country.

- Use **The Canadian Press Stylebook** with the exceptions noted in our corporate style guide. For spelling, use **CP Caps and Spelling** or the **Canadian Oxford Dictionary**.
 - Refer to our online **Corporate Identity Guide** for detailed information on how to use our corporate logos, fonts, colours, trademarks and company names. It also contains information on product advertising and marketing guidelines for The Co-operators, Group marketing, and CUMIS brand standards and more.
 - There are rules about how to write The Co-operators in English and French. For example, we can't break it over a line or use it as a possessive. The full rules are in the Corporate Identity Guide.
 - **User Experience (UX) writing** guides users within a product and helps them interact with it. As with other types of writing in the organization, UX writing follows the brand voice and style guidelines outlined in our corporate style guide.
 - Please visit <https://source-cooperators.ca/home/about-us/brand-guidelines/style-guide> before writing copy for an interface.

Best practices for writing copy

Here are a few quick tips for writing for The Co-operators online. If you have questions, consult with the content writer on your project or Luisa Lago. Ensure all content uses plain language. Our brand voice is to be **authentic, positive, and conversational**. We use this same voice in our online content by being clear in our instructions, descriptions, messages, notifications, etc.

- Use sentence case for headings and labels. Start with a capital, and cap only proper names.
- Use descriptive, long-tail hyperlinks. Do NOT use “click here” or its variants. Use keywords and make links longer than two or three words to make them easier to see and click.
- Consider translations when determining spacing. French is typically 30% longer than English.
- Keep visually and physically impaired readers in mind. Think of adding descriptive screen reader text wherever possible. Avoid putting important content in images; if it's unavoidable, use descriptive ALT tags. Avoid making clickable targets too small or using low-contrast designs.
- Use abbreviations and acronyms sparingly. If they're necessary for space, ensure the first mention is spelled out or otherwise clear to the reader. Also avoid repeating elements of the acronym; for example, use “VIN” not “VIN number”.
- Check for commonly misused and misspelled words. For example: client (not customer), coverage (not coverages), sign in (not login, log on, etc.) auto liability slip (not pink slip).
- Always spell our company name correctly. It's always “The Co-operators” with the caps and hyphen, except in “Your Co-operators policy”. Avoid using “co-op”, “Cooperators” and other variations, especially in client-facing material. always keep “The” and “Co-operators” together on one line.

Why tokens?

About tokens

Tokens are variables where CSS values can be stored. The value in this CSS variable—say, a hexcode for colour or a pixel value for spacing—are set in one place, a style sheet, and the token can be used everywhere. That means that changes can be made to the look and feel of the website in one place, and then those changes are distributed everywhere automatically via the tokens. Tokens are a best practice for web development and are now considered standard.

Using tokens makes the work of designers more consistent and saves development time; rather than taking 20 minutes to coding a title every single place it's used, that title can be coded once and distributed. Tokens shave sprint points off every story and reduce needless duplicate work. They also drive consistency across the user experience, because the variables controlling look and feel are set in one place, not every time they're used.

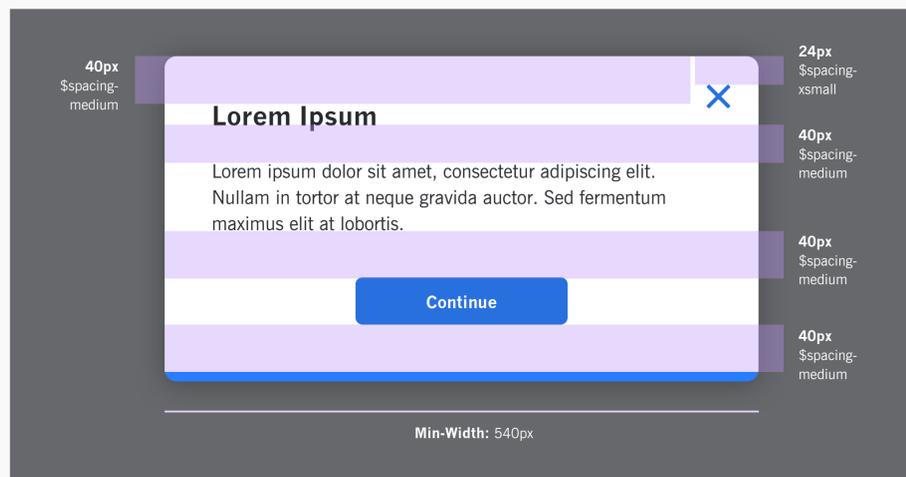
Re-using tokens components from a centralized style sheet and library also reduces QA time; components are not re-written with each use, so they don't need to be re-tested.

Co-operators is changing its branding in fall 2020. These changes will affect every component on Cooperators.ca and within CX. If tokens are used, the updates can be made in one place. If CSS is written manually for each component, then the updates will need to be made for each component, multiplying developer time by a factor of at least ten and costing significantly more.

Using tokens as a designer

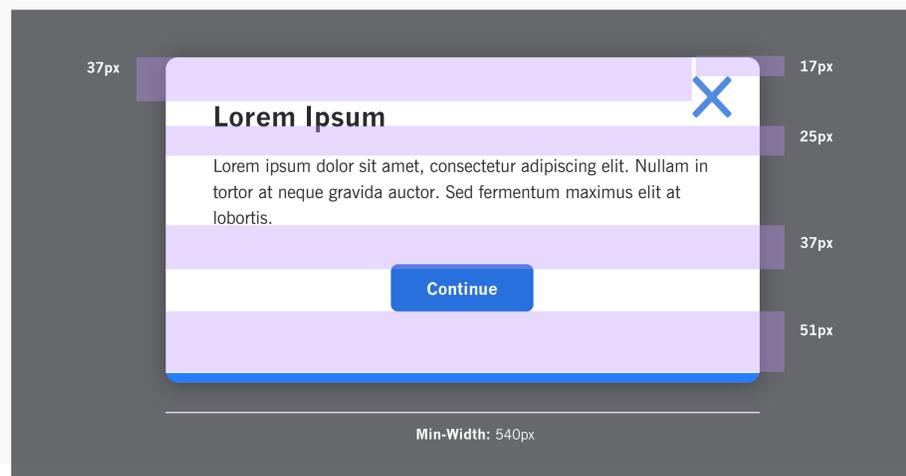
Designers are accountable to using spacing, fonts, and colours as defined in this design system. If we create values for developers that do not align to the system, developers can't use tokens to implement them in the CSS. For example, if we create a 9px by 9px box shadow, or a margin of 9 pixels with a background colour not found in this system, those values will have to be manually set by developers, defeating the purpose of tokens and making the upcoming branding changes difficult.

A component that will require less manual CSS work because it follows the tokens



Modal Title	Modal Body Copy	Close 'X' Icon	Shadow
Font: \$font-heading-medium	Font: \$colour-brand-black	Colour: \$colour-ui-graphics	Colour: \$colour-brand-black
Font Colour: \$colour-brand-black	Font Colour: \$font-body-medium	Size: \$border-radius-card	@30%
		Y: 2px	Blur: \$spacing-xsmall

A component requiring more manual work and testing time because it doesn't follow the design tokens



Modal Title	Modal Body Copy	Close 'X' Icon	Shadow
Font-size: 19px;	Font-size: 15px;	Colour: #508BE4;	Colour: #000000 @30%
Font Colour: #000000	Font Colour: #000000	Size: 33px x 35px	Y: 2px
			Blur: 20px

Using tokens as a developer

Tokens are building blocks, or the atoms of a design. In this design system, we've laid out token names that will be able to stay the same with the branding changes. Working toward one centralized component library and living design system across product development teams, developers can help save time by using the names as they're set out here.

CSS that has to be written manually for each component, and will have to be updated manually within each component with the branding changes

```
&__item {
  flex: none;
  box-shadow: 0 8px 20px 0 fade-out($colour-
primary-03,0.84);
  background-color: $color-white;
  border-radius: 10px;
  min-height: 139px;
  width: 378px;
  padding: 29px 24px 20px;
  margin: 0 24px 40px 0;
}
```

Here, some tokens are used, but they don't match those defined in this design system and are likely different than the ones used across other product teams, creating an obstacle to a centralized design system and reduced time spent on for accessibility testing.

Tokens aren't used for spacing, which means the component will have to be updated line by line with the styling changes coming in the fall. That's 4x more work

CSS using tokens, that does not have to be rewritten for each component with the branding changes

```
&__item {
  flex: none;
  box-shadow: 0 $px $spacing-xsmall 0 fade-out($colour-
brand-white, .30);
  background-color: $color-brand-white;
  border-radius: $border-radius-card;
  min-height: 139px;
  width: 378px;
  padding: $spacing-xsmall, $spacing-xsmall $spacing-2xsmall;
  margin: 0 $spacing-xsmall $spacing-medium 0;
}
```

Here, more tokens are used, and they align to the exact names defined in this system.

Only two lines of CSS will need to be rewritten when a style update needs to be completed, as opposed to 8 lines that need to be updated in the example to the left.

Font Tokens

Line height

Base	Line-height: 1.5em <code>\$type-lineheight-base</code>
Big	Line-height: 2em <code>\$type-lineheight-big</code>

Headings

xxsmall heading	xxsmall-heading Type: News Gothic Size: 14px Line height: 18px Weight: Bold Letter-spacing: 0px <code>\$font-heading-xxsmall</code>
xsmall heading	xsmall-heading Type: News Gothic Size: 16px Line height: 22px Weight: Bold Letter-spacing: 0px <code>\$font-xsmall-heading</code>
small heading	small-heading Type: News Gothic Size: 18px Line height: 24px Weight: Bold Letter-spacing: 0px <code>\$font-small-heading</code>
medium heading	medium-heading Type: News Gothic Size: 22px Line height: 28px Weight: Bold Letter-spacing: 0px <code>\$font-medium-heading</code>
large heading	large-heading Type: News Gothic Size: 28px Line height: 32px Weight: Bold Letter-spacing: 0px <code>\$font-large-heading</code>
xlarge heading	xlarge-heading Type: News Gothic Size: 36px Line height: 40px Weight: Bold Letter-spacing: 0px <code>\$font-xlarge-heading</code>

Body

body xsmall. Commonly used for legal content.	body-xsmall Type: News Gothic Size: 12px Line height: 16px Weight: Regular Letter-spacing: 0px <code>\$font-body-xsmall</code>
body-small. Commonly used in components and notifications.	body-small Type: News Gothic Size: 14px Line height: 18px Weight: Regular Letter-spacing: 0px <code>\$font-body-small</code>
body-medium. Commonly used in components, paragraphs and descriptive texts.	body-medium Type: News Gothic Size: 16px Line height: 22px Weight: Regular Letter-spacing: 0px <code>\$font-body-medium</code>
body-large	body-large Type: News Gothic Size: 18px Line height: 24px Weight: Regular Letter-spacing: 0px <code>\$font-body-large</code>

Colour Tokens

Primary Colours

Token	Primary Usage	Value
<code>\$colour-coop-white</code>	Used for light backgrounds or inverted states on dark backgrounds	Coop White - #FFFFFF
<code>\$colour-brand-blue-10</code>	Used in containers and graphic elements	Brand Blue 10 - #E9F2FF
<code>\$colour-brand-blue-11</code>	Used in containers and graphic elements	Brand Blue 11 - #D4E5FF
<code>\$colour-ui-graphics</code>	Used for emphasis in graphics and form fields	UI Graphics - #267DFF
<code>\$colour-brand-blue</code>	Primary Button	Brand Blue - #2770DD
<code>\$colour-dark-blue</code>	Used in containers and graphic elements	Dark Blue - #1753AC
<code>\$colour-marketing-blue</code>	Used in containers and graphic elements	Marketing Blue - #5078AA
<code>\$colour-brand-black</code>	Often used in type and headings	Brand Black - #282A2E
<code>\$colour-brand-grey</code>	Often used in lines and form fields	Brand Grey - #6D747F
<code>\$colour-medium-grey</code>	Often used in form fields	Medium Grey - #91939C

Feedback

Token	Primary Usage	Value
<code>\$colour-error</code>	For errors or failures	Red - #C92929
<code>\$colour-warning</code>	For warnings	Yellow - #F6B014
<code>\$colour-success</code>	For successes, saves, good news	Green - #24A148
<code>\$colour-disabled</code>	For disabled states used in labels and input boxes etc.	Disabled - #F4F4F4
<code>\$colour-inactive</code>	Commonly used for the inactive state of progress bars.	Inactive - #E2E3E5

Grid and breakpoints

Breakpoints

Token	Breakpoint	Specs	For
<code>\$breakpoint-small</code>	Mobile 0 - 768px	Grid Width: 359px Margins (on either side): 8px Number of Columns: 4 Columns: 78px Column Gutters: 16px	Artboard Size: 375px Grid Size: 359px + 8px outer margins
<code>\$breakpoint-medium</code>	Tablet 768 - 992px	Grid Width: 720px Margins (on either side): 24px Number of Columns: 12 Columns: 38px Column Gutters: 24px	Artboard Size: 768px Grid Size: 720px + 24px outer margins
<code>\$breakpoint-large</code>	Desktop 992px +	Grid Width: 1280px Margins (on either side): 30px Number of Columns: 12 Columns: 64px Column Gutters: 32px	Artboard Size: 1440px Grid Size: 1280px + 30px outer margins
	XL - Desktop We are not designing or testing at this breakpoint yet in CX, and we're confirming whether or not to pursue this fourth breakpoint.		

Desktop – 1280px | 32G/64C

Tablet – 720px | 24G/38C

Mobile – 359px | 16G/78C

Animation tokens

Basics

Token	Primary Usage	Value
<code>\$animate-transition-duration</code>	Used for buttons and cards, changing states	<code>transition-duration: .3s;</code>
<code>\$animate-transition-timing</code>	Default setting for most animation in most browsers	<code>transition-timing-function: ease;</code>

Spacing & Other Tokens

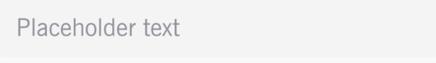
Spacing

Token	px	Example
<code>\$spacing-3xsmall</code>	8px	
<code>\$spacing-2xsmall</code>	16px	
<code>\$spacing-xsmall</code>	24px	
<code>\$spacing-small</code>	32px	
<code>\$spacing-medium</code>	40px	
<code>\$spacing-large</code>	48px	
<code>\$spacing-xlarge</code>	80px	
<code>\$spacing-2xlarge</code>	120px	

Other Tokens

Token	px	Example
<code>\$border-width</code>	1px	
<code>\$border-width-2</code>	2px	
<code>\$border-width-8</code>	8px	
<code>\$border-radius</code>	3px radius	
<code>\$border-radius-button</code>	6px radius	
<code>\$border-radius-card</code>	10px radius	
<code>\$shadow-small</code>	Colour: <code>\$colour-ui-graphics</code> , 50% alpha Blur: 4px	
<code>\$shadow-dropdown</code>	Colour: <code>\$colour-dark-blue</code> , 20% alpha Blur: 8px, y-axis: 2px	
<code>\$shadow-big</code>	Colour: <code>\$colour-dark-blue</code> , 16% alpha Blur: 20px	
<code>\$shadow-modal</code>	Colour: <code>\$colour-brand-black</code> , 20% alpha Blur: 20px, y-axis: 2px	

Text Fields

State	Example	Tokens Used
<p>:enabled width: can vary height: 40px</p>	<p>Label</p>  <p>Label</p>  <p>10px</p> <p>10px</p>	<p>Label</p> <p>Font: \$font-heading-xsmall Colour: \$colour-brand-black</p> <p>Input Box</p> <p>Border: \$border-width, \$border-radius Colour: \$colour-medium-grey</p> <p>Placeholder Text</p> <p>Font: \$font-heading-xsmall Colour: \$colour-grey-medium</p>
<p>:hover</p>	<p>Label</p> 	
<p>:focus</p>	<p>Label</p> 	<p>Input Box</p> <p>Colour: \$colour-ui-graphics</p>
<p>:active</p>	<p>Label</p> 	<p>Input Box</p> <p>Colour: \$colour-ui-graphics</p> <p>Input Text</p> <p>Colour: \$colour-brand-black</p>
<p>Filled</p>	<p>Label</p> 	<p>Input Text</p> <p>Colour: \$colour-brand-black</p>
<p>Helper Text</p>	<p>Label</p>  <p>Helper text here</p> <p>Label</p>  <p>8px (\$spacing-3xsmall)</p> <p>8px (\$spacing-3xsmall)</p>	<p>Helper Text</p> <p>Font: \$font-heading-xsmall Colour: \$colour-brand-grey</p>
<p>:error</p>	<p>Label</p>  <p>❗ Error message here</p> <p>Label</p>  <p>❗ Error message here</p> <p>8px (\$spacing-3xsmall)</p> <p>8px (\$spacing-3xsmall)</p> <p>Error Icon Size: 16 x 16px</p>	<p>Input Box</p> <p>Colour: \$colour-error</p> <p>Error Text</p> <p>Font: \$font-heading-xsmall Colour: \$colour-error</p>
<p>with Tooltip</p>	<p>Label ⓘ</p>  <p>8px (\$spacing-3xsmall)</p> <p>Label ⓘ</p>  <p>Tooltip Size: 16 x 16px</p>	<p>Tooltip Icon</p> <p>Colour: \$colour-brand-black</p>
<p>:disabled</p>	<p>Label</p> 	<p>Label</p> <p>Colour: \$colour-brand-grey</p> <p>Input Box</p> <p>Colour: \$colour-disabled</p> <p>Input Text</p> <p>Colour: \$colour-grey-medium</p>

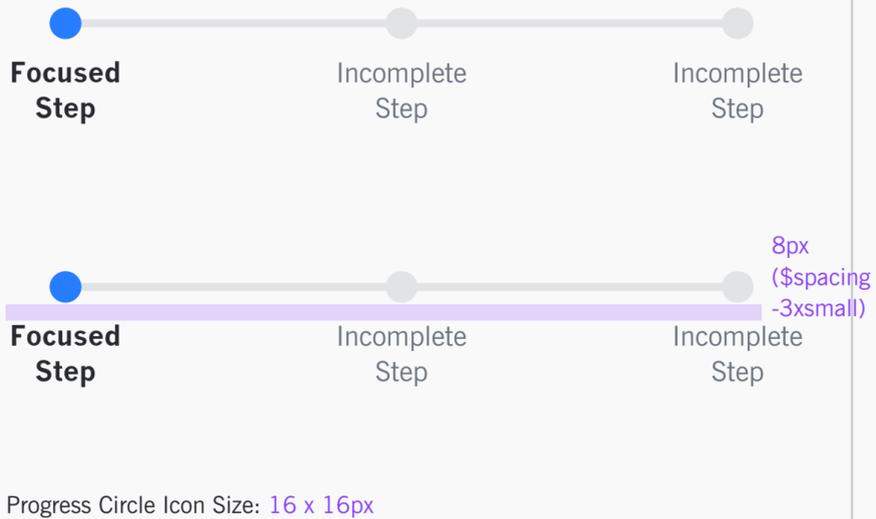
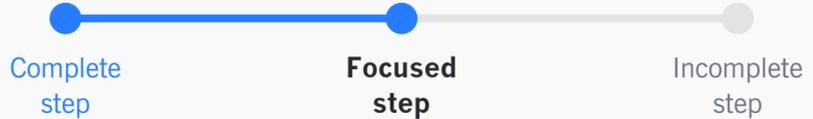
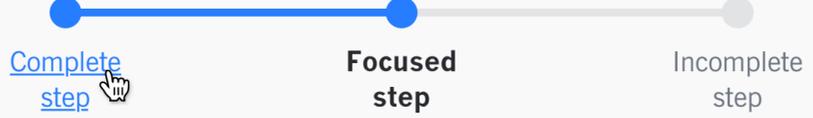
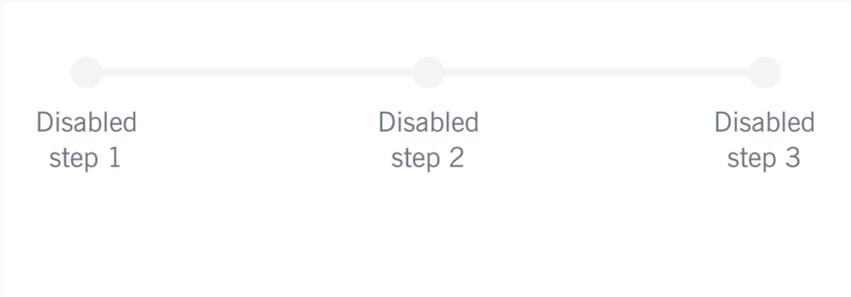
Checkbox

State	Example	Tokens Used
:default Size: 16 x 16px	<input type="checkbox"/> Label <input type="checkbox"/> Label 14px	Label Font: \$font-body-medium Colour: \$colour-brand-black Checkbox Border: \$border-width-2, \$border-radius Border Colour: \$colour-medium-grey Fill: \$colour-coop-white
:focused	<input type="checkbox"/> Label	Checkbox Border: \$border-width-2, \$border-radius Border Colour: \$colour-ui-graphics Shadow: \$shadow-small
:disabled	<input type="checkbox"/> Label	Checkbox Border Colour: \$colour-medium-grey, 50% alpha Fill: \$colour-disabled
:error	<input type="checkbox"/> Label	Checkbox Border Colour: \$colour-error Fill: \$colour-error, 20% alpha
:selected	<input checked="" type="checkbox"/> Label	Checkbox Border Colour: \$colour-ui-graphics Checkmark Colour: \$colour-ui-graphics
:selected/focus	<input checked="" type="checkbox"/> Label	Checkbox Border Colour: \$colour-ui-graphics Shadow: \$shadow-small
:selected disabled	<input checked="" type="checkbox"/> Label	Checkbox Border Colour: \$colour-medium-grey, 50% alpha Fill: \$colour-disabled Checkmark Colour: \$colour-medium-grey, 70% alpha
:selected error	<input checked="" type="checkbox"/> Label	Checkbox Border Colour: \$colour-error Fill: \$colour-error, 20% alpha Checkmark Colour: \$colour-error

Radio Button

State	Example	Tokens Used
:default Size: 16 x 16px	 	Label Font: \$font-body-medium Colour: \$colour-brand-black Radio Border: \$border-width-2 Border Colour: \$colour-medium-grey Fill: \$colour-coop-white
:focused		Radio Border Colour: \$colour-ui-graphics Shadow: \$shadow-small
:disabled		Radio Border Colour: \$colour-medium-grey, 50% alpha Fill: \$colour-disabled
:selected		Radio Border Colour: \$colour-ui-graphics Fill: \$colour-coop-white Inner Circle Fill: \$colour-ui-graphics
:selected/focus		Radio Border Colour: \$colour-ui-graphics Fill: \$colour-coop-white Shadow: \$shadow-small Inner Circle Fill: \$colour-ui-graphics
:selected disabled		Radio Border Colour: \$colour-medium-grey, 50% alpha Fill: \$colour-disabled Inner Circle Fill: \$colour-medium-grey, 50% alpha

Progress Indicator

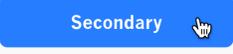
State	Example	Tokens Used
:default	 <p>Progress Circle Icon Size: 16 x 16px</p>	<p>Bar</p> <p>Colour: \$colour-inactive</p> <hr/> <p>Focused Step</p> <p>Font: \$font-heading-xxsmall Font Colour: \$colour-brand-black Icon: \$colour-ui-graphics</p> <p>Incomplete Step</p> <p>Font: \$font-body-small Font Colour: \$colour-brand-grey Icon: \$colour-inactive</p>
:step completed		<p>Bar</p> <p>Completed Step Colour: \$colour-ui-graphics</p> <hr/> <p>Complete Step</p> <p>Font: \$font-body-small Font Colour: \$colour-ui-graphics Icon: \$colour-ui-graphics</p>
:hover		
:disabled		<p>Disabled Bar & Icon</p> <p>Colour: \$colour-disabled</p> <hr/> <p>Disabled Step</p> <p>Font: \$font-body-small Font Colour: \$colour-brand-grey Icon: \$colour-disabled</p>

Buttons

Primary Button

State	Example	Tokens Used
:default (height: 40px)	 Min Width: 180px	Colour: \$colour-brand-blue Border: \$border-radius-button Font: \$font-heading-xxsmall Font Colour: \$colour-coop-white
large button	  32px (\$spacing-small) 32px (\$spacing-small)	Colour: \$colour-medium-grey
:hover		Colour: \$colour-dark-blue
:clicked/ pressed		Colour: \$colour-dark-blue, 60% alpha Font Colour: \$colour-coop-white, 60% alpha
:disabled		Colour: \$colour-medium-grey

Secondary Button

State	Example	Tokens Used
:default (height: 40px)		Border: \$border-width, \$border-radius-button Border Colour: \$colour-brand-blue Font: \$font-heading-xxsmall Font Colour: \$colour-brand-blue
:hover		Colour: \$colour-ui-graphics
:clicked/ pressed		Colour: \$colour-ui-graphics, 60% alpha Font Colour: \$colour-coop-white, 60% alpha
:disabled		Colour: \$colour-medium-grey Font Colour: \$colour-medium-grey

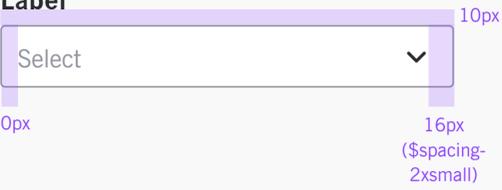
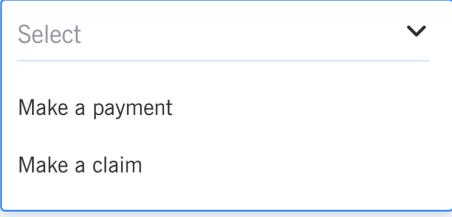
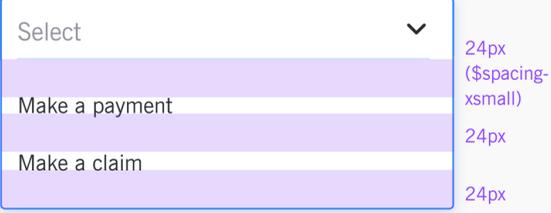
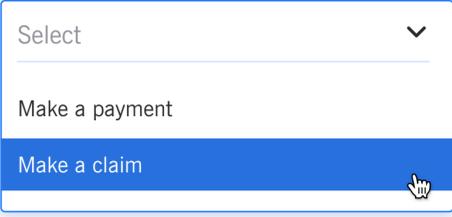
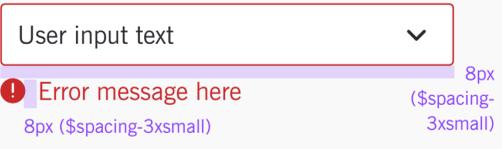
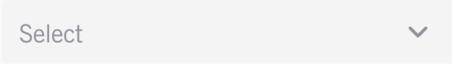
Quick Link Button

State	Example	Tokens Used
:default (height: 40px)	  16px (\$spacing-2xsmall)	Font: \$font-heading-small Font Colour: \$colour-brand-blue Icon: \$colour-brand-blue-11 Icon Arrow: \$colour-ui-graphics
:hover		Underline: \$colour-ui-graphics, \$border-width-2 Hover Ring: #7DB1FF
:clicked/ pressed		60% alpha
:disabled		Font Colour: \$colour-medium-grey Icon: \$colour-medium-grey Icon Arrow: \$colour-coop-white

Text Link

State	Example	Tokens Used
:default		Font: \$font-heading-small Font Colour: \$colour-brand-blue Icon: \$colour-brand-blue-11 Icon Arrow: \$colour-ui-graphics
:hover		Underline: \$colour-ui-graphics, \$border-width-2 Hover Ring: #7DB1FF
:clicked/ pressed		60% alpha

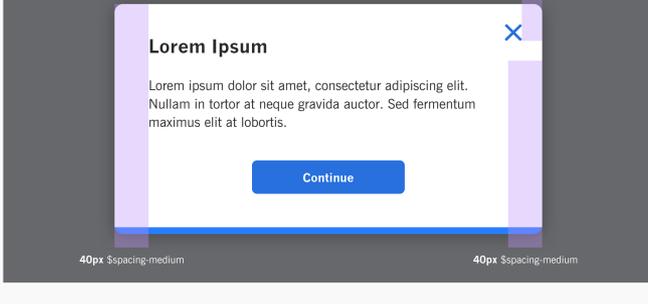
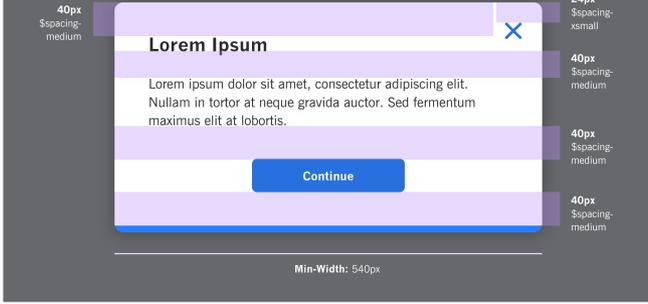
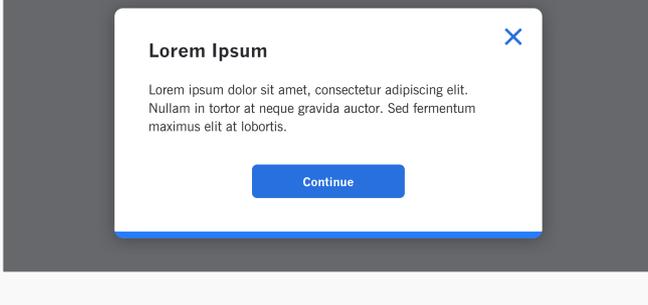
Dropdowns

State	Example	Tokens Used
<p>:enabled width: can vary height: 40px</p>	<p>Label</p>  <p>Label</p>  <p>10px</p> <p>16px (\$spacing-2xsmall)</p> <p>Dropdown Icon Size: 16 x 16px</p>	<p>Label</p> <p>Font: \$font-heading-xsmall Colour: \$colour-brand-black</p> <p>Input Box</p> <p>Border: \$border-width, \$border-radius Border Colour: \$colour-medium-grey</p> <p>Placeholder Text</p> <p>Font: \$font-heading-xsmall Colour: \$colour-grey-medium</p> <p>Dropdown Icon</p> <p>Colour: \$colour-brand-black</p>
<p>:focus</p>	<p>Label</p> 	<p>Input Box</p> <p>Border Colour: \$colour-ui-graphics</p>
<p>:active</p>	<p>Label</p>  <p>Make a payment</p> <p>Make a claim</p> <p>Label</p>  <p>24px (\$spacing-xsmall)</p> <p>24px</p> <p>24px</p> <p>Label</p> 	<p>Input Box</p> <p>Border Colour: \$colour-ui-graphics Shadow: \$shadow-modal Separator Line: \$colour-brand-blue-11</p> <p>Selections Text</p> <p>Font: \$font-body-small Colour: \$colour-brand-black</p> <p>Selection Hover</p> <p>Colour: \$colour-brand-blue</p>
<p>Filled</p>	<p>Label</p> 	<p>Input Text</p> <p>Colour: \$colour-brand-black</p>
<p>:error</p>	<p>Label</p>  <p>! Error message here</p> <p>Label</p>  <p>! Error message here</p> <p>8px (\$spacing-3xsmall)</p> <p>Error Icon Size: 16 x 16px</p>	<p>Input Box</p> <p>Border Colour: \$colour-error</p> <p>Error Text</p> <p>Font: \$font-heading-xsmall Colour: \$colour-error</p>
<p>:disabled</p>	<p>Label</p> 	<p>Label</p> <p>Colour: \$colour-brand-grey</p> <p>Input Box</p> <p>Border Colour: \$colour-disabled</p> <p>Input Text</p> <p>Colour: \$colour-grey-medium</p> <p>Dropdown Icon</p> <p>Colour: \$colour-grey-medium</p>

Modals

Basic Modal

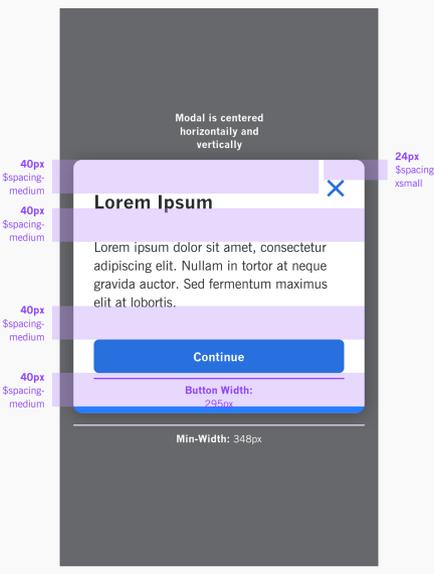
Desktop Example



Tokens Used

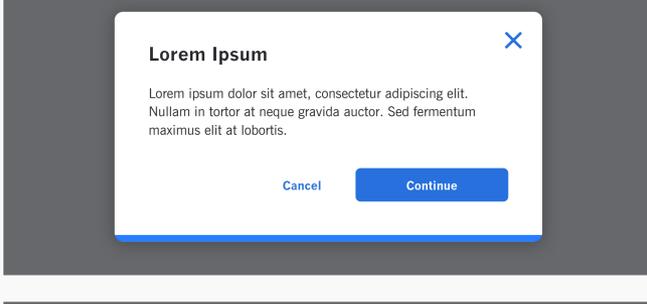
- Background**
Colour: \$colour-brand-black, 70% alpha
- Modal Box**
Fill Colour: \$colour-coop-white
- Border Radius:**
\$border-radius-card
- Bottom Border:**
\$border-width-8
- Bottom Border Colour:**
\$colour-ui-graphics
- Modal Title**
Font: \$font-heading-medium
- Font Colour:**
\$colour-brand-black
- Modal Body Copy**
Font: \$colour-coop-white
- Font Colour:**
\$font-body-medium
- Close 'X' Icon**
Colour: \$colour-ui-graphics
- Size:**
\$border-radius-card

Mobile Example

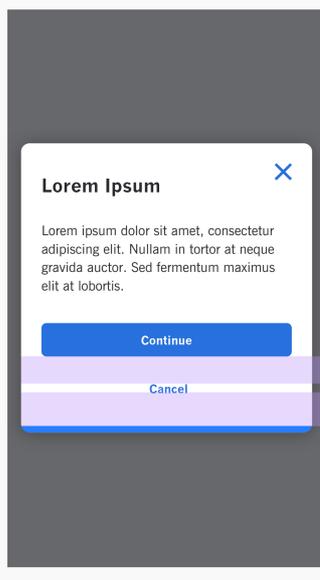
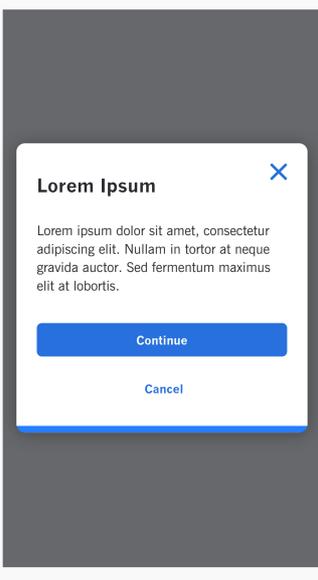


Modal with 2 buttons

Desktop Example

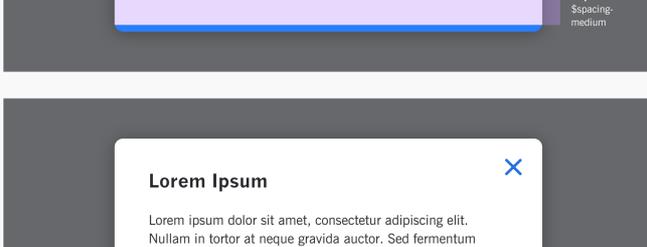
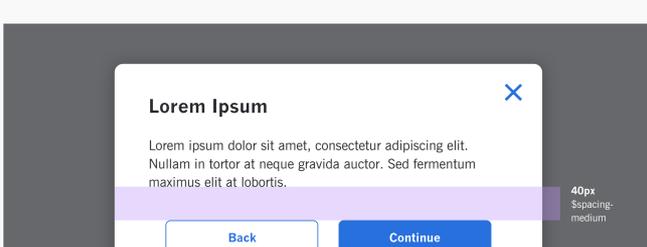
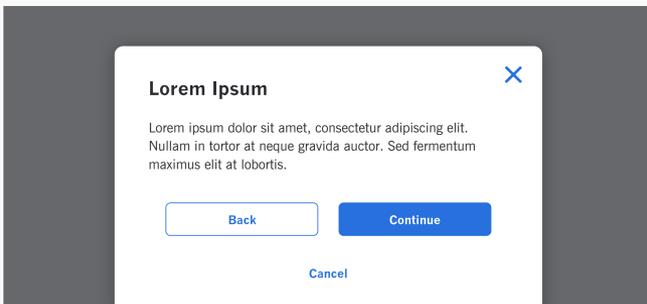


Mobile Example



Modal with 3 buttons

Desktop Example



Mobile Example

