

Components

Behavior

Behavior is the table's dynamic reactions to user interactions or system events.

Sticky Row/Column

Remains visible and fixed in its position on the screen, even when a user scrolls through the rest of the table. This feature is often used for headers or important data to ensure they're always accessible as the user navigates the table. Play prototype to see sticky behavior. Sticky behavior is implemented through property in code.

Column Header	Column Header	Column Header	Column Header	Column Header
Detail	Detail	Detail	Detail	Detail
Detail	Detail	Detail	Detail	Detail
Detail	Detail	Detail	Detail	Detail
Detail	Detail	Detail	Detail	Detail

- Row.** The header row is sticky when scrolling up and down to allow users to see column titles or essential header information are constantly accessible.

Column Header	Column Header	↑	Column Header	Column Header	Column Header
Detail	Detail		Detail	Detail	Detail
Detail	Detail		Detail	Detail	Detail
Detail	Detail		Detail	Detail	Detail
Detail	Detail		Detail	Detail	Detail
Detail	Detail		Detail	Detail	Detail
Detail	Detail		Detail	Detail	Detail
Detail	Detail		Detail	Detail	Detail
Detail	Detail		Detail	Detail	Detail

- Column.** A column or set of columns are sticky when scrolling left and right to allow users to see vast datasets while still maintaining access to related information. A dark grey border on the right indicates the edge to emphasize the difference between the sticky column and the cells behind it.

Vertical Scrolling

Allows users to move up and down within a table to view content that is not currently visible on the screen. This behavior is essential for tables with many rows that cannot all be displayed at once. It uses infinite scroll.

Column Header	Column Header	Column Header	Column Header	Column Header
Detail	Detail	Detail	Detail	Detail
Detail	Detail	Detail	Detail	Detail
Detail	Detail	Detail	Detail	Detail
Detail	Detail	Detail	Detail	Detail
Detail	Detail	Detail	Detail	Detail
Detail	Detail	Detail	Detail	Detail
Detail	Detail	Detail	Detail	Detail

Horizontal Scrolling

Lets users move left and right within a table to see content that extends beyond the screen's width. This is particularly useful for tables with a large number of columns. It uses infinite scroll.

Column Header	Column Header	Column Header	Column Header	Column Header	Column Header
Detail	Detail	Detail	Detail	Detail	Detail
Detail	Detail	Detail	Detail	Detail	Detail
Detail	Detail	Detail	Detail	Detail	Detail
Detail	Detail	Detail	Detail	Detail	Detail
Detail	Detail	Detail	Detail	Detail	Detail
Detail	Detail	Detail	Detail	Detail	Detail
Detail	Detail	Detail	Detail	Detail	Detail
Detail	Detail	Detail	Detail	Detail	Detail

Virtual Scrolling

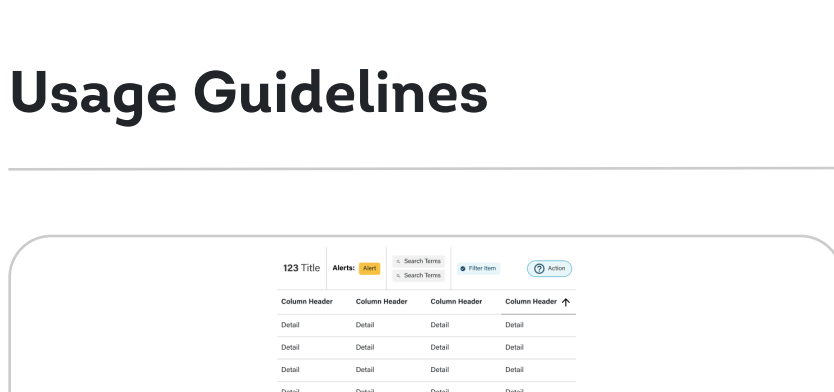
Technique where only the visible rows or columns of a table are rendered in the DOM (Document Object Model) at any given time. As the user scrolls, off-screen rows or columns are removed from the DOM and new ones are added, optimizing performance for large data sets.

<By assigned dev>

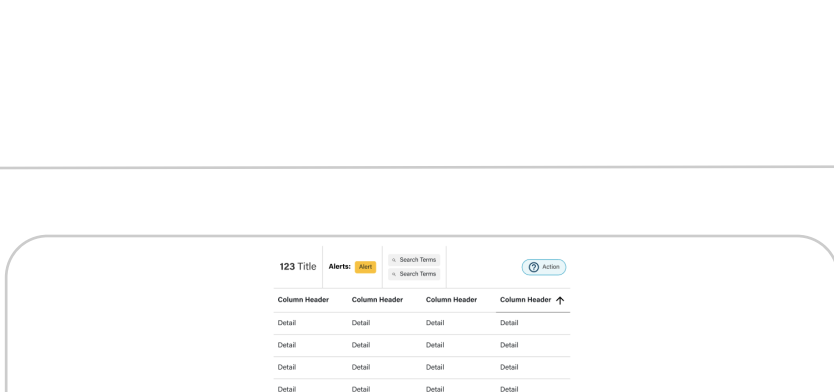
Accessibility

- ✔ Has a contrast ratio of at least 4.5:1 for small text (WCAG 2.0 1.4.3)
- ✔ Provide alternative text.
- ✔ Able to navigate through keyboard (2.1.1 Keyboard Operability)

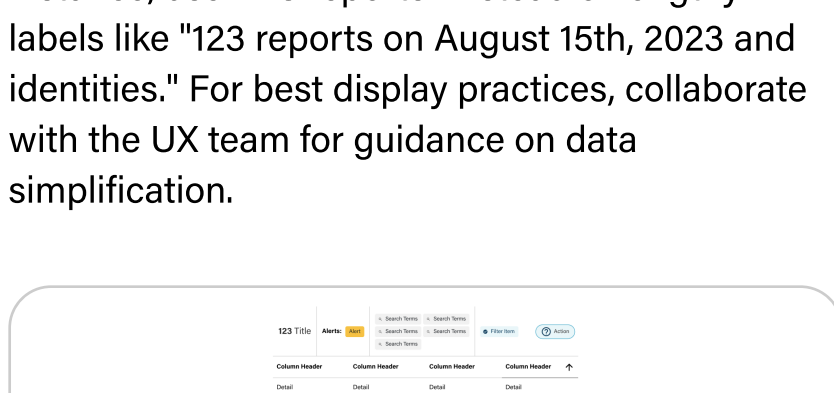
Usage Guidelines



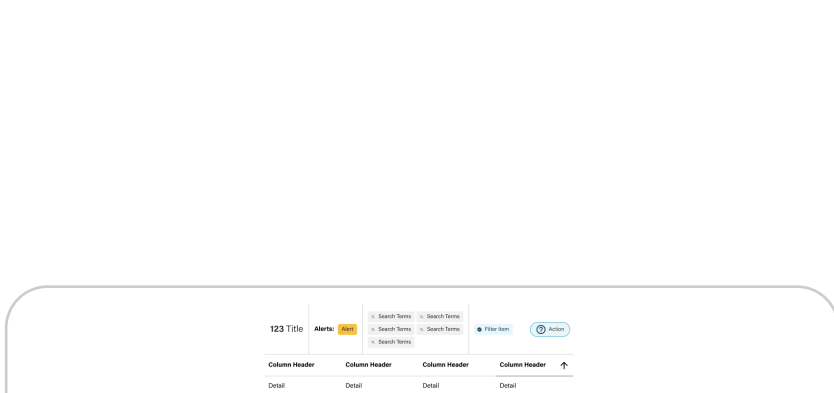
- ✔ **Do**
Table controls (e.g., title, alerts, search terms) have no predefined width constraints. Product teams should set appropriate boundaries, ensuring titles are concise and easily understandable. For instance, use "123 reports" instead of lengthy labels like "123 reports on August 15th, 2023 and identities." For best display practices, collaborate with the UX team for guidance on data simplification.



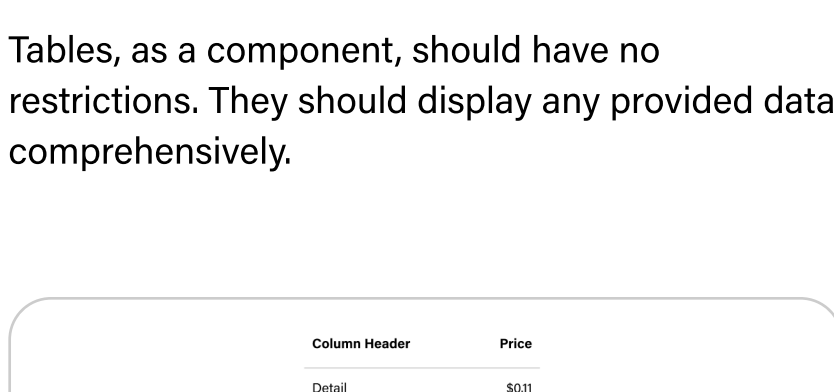
- ✘ **Don't**
Avoid setting minimum/maximum width for table controls.



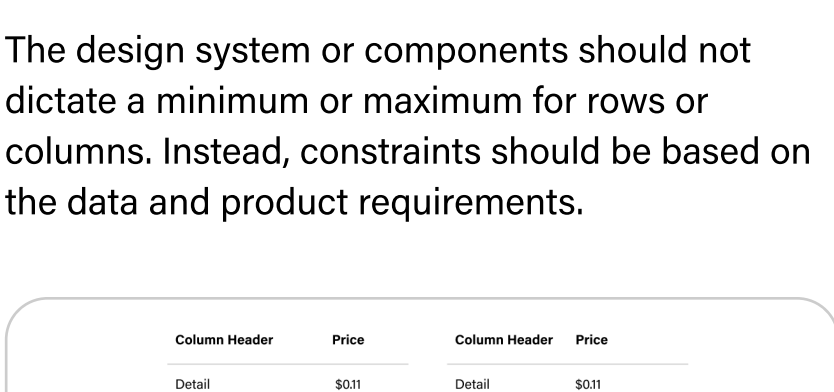
- ✔ **Do**
Tables, as a component, should have no restrictions. They should display any provided data comprehensively.



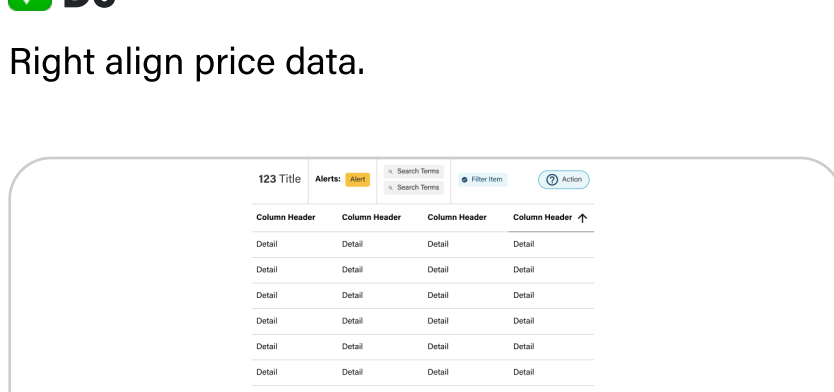
- ✘ **Don't**
The design system or components should not dictate a minimum or maximum for rows or columns. Instead, constraints should be based on the data and product requirements.



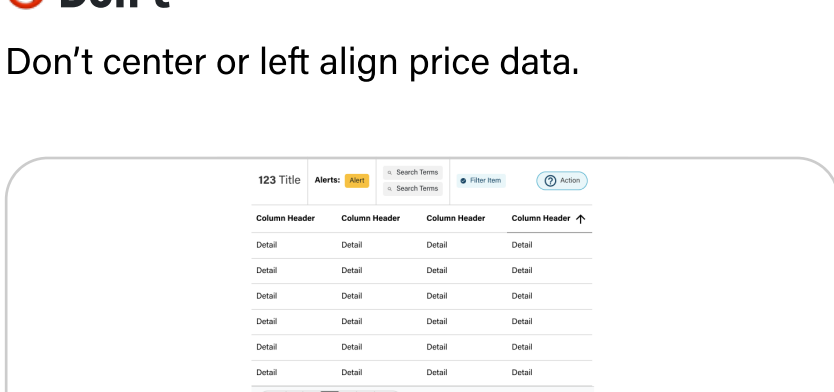
- ✔ **Do**
Right align price data.



- ✘ **Don't**
Don't center or left align price data.



- ✔ **Do**
Always implement infinite scrolling for data tables.



- ✘ **Don't**
Avoid using pagination. While it exists due to documentation practices, it's not intended for actual use.

*(End of Phase 4)

Changelogs:

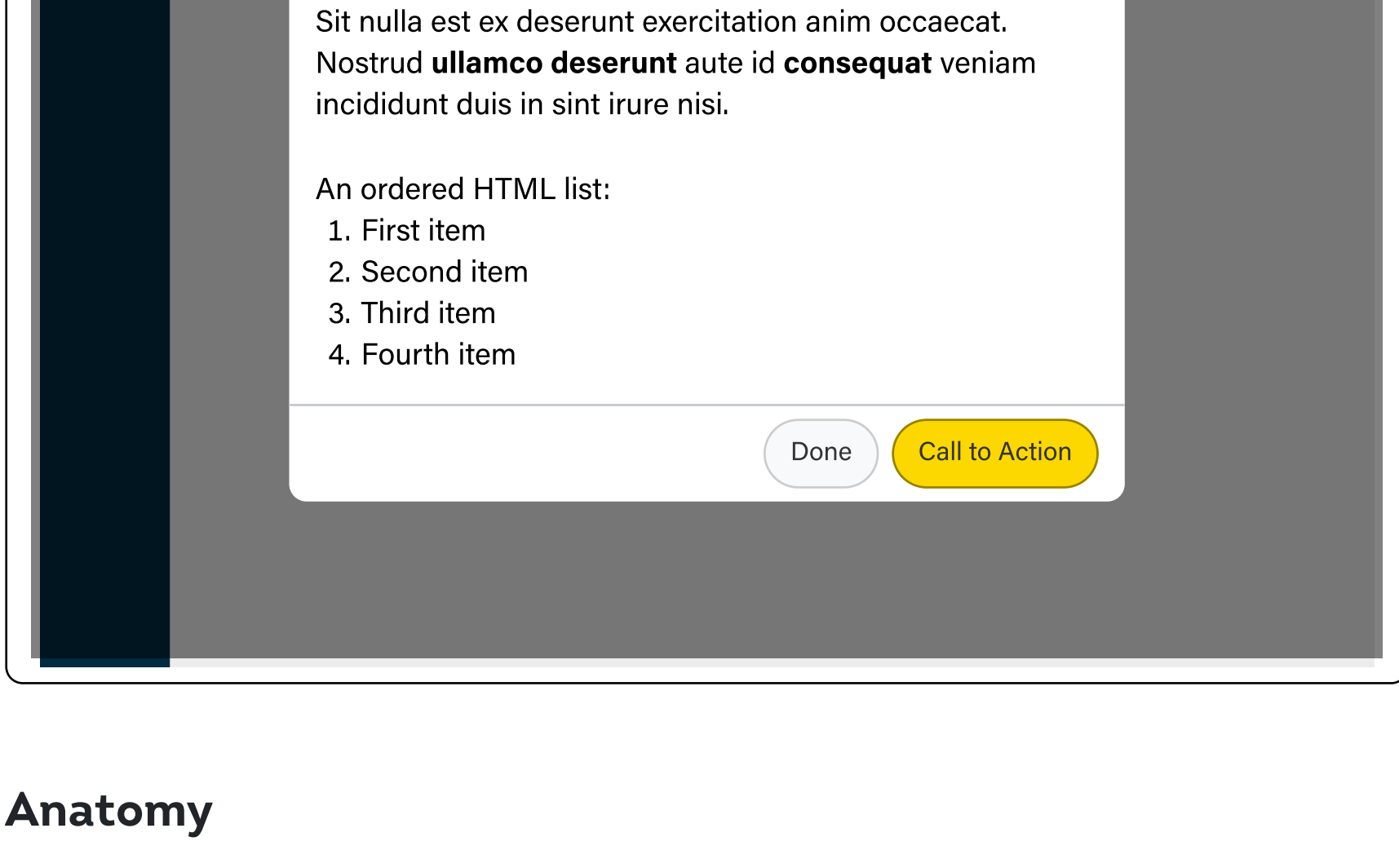
- 11/21/2023 - For default padding - left right: spacing-3 top bottom: spacing 4
For comfortable padding: left right: spacing-4 top bottom: spacing 5 (EI)

Modal

- Overview
- Anatomy
- Props
- Sizes
- Behaviors
- Accessibility
- Usage Guidelines

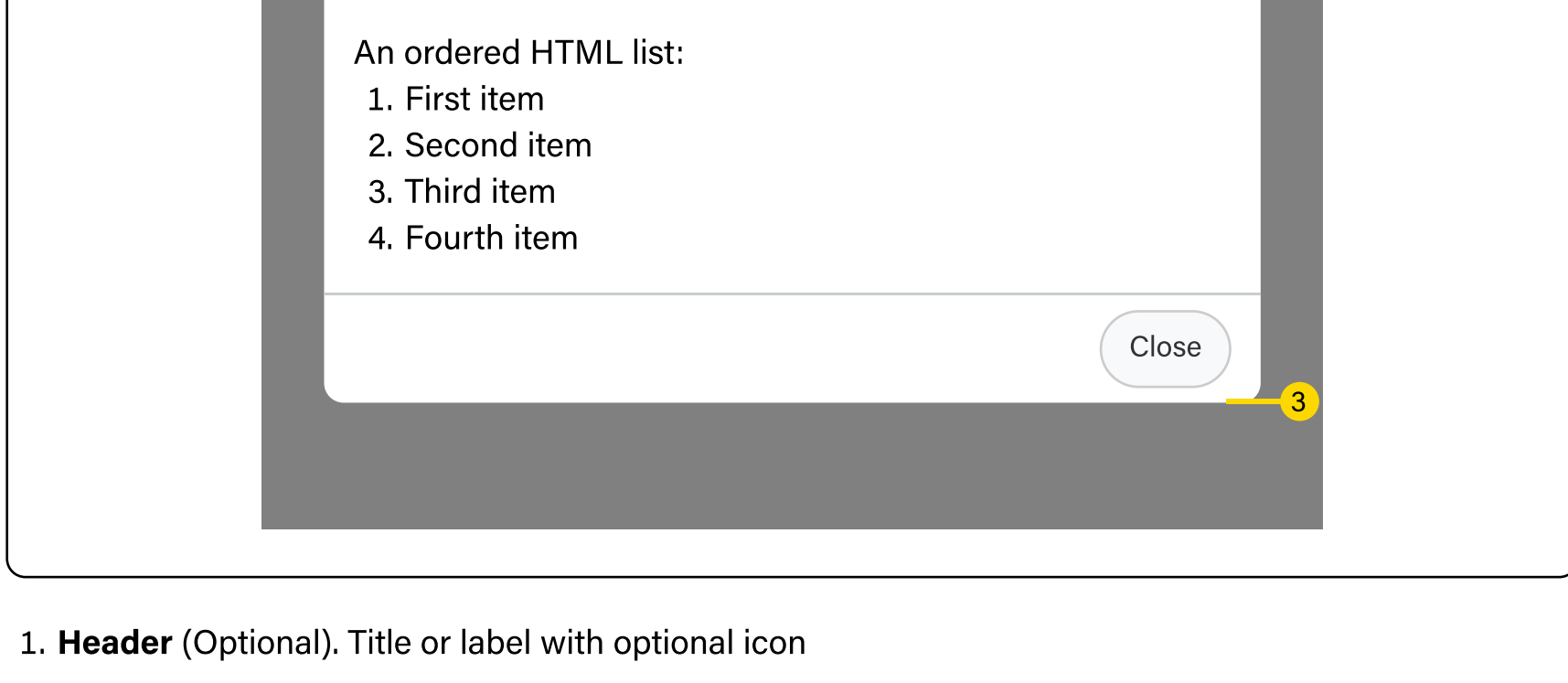
Overview

Dialogs that interrupts a user's page experience when presenting secondary content overlayed above the main content.



Anatomy

A modal at the very least has a fixed style header with the body and footer having a free form.



1. **Header** (Optional). Title or label with optional icon
2. **Body**. Message text containing information or controls
3. **Footer** (Optional). Usually a button group to process a task or complete an action. Should have at least one button for acknowledging/dismissing the modal.
4. **X Icon**. Use to dismiss the modal
5. **Backdrop**. Dark transparent overlay slightly covering the contents of the current page

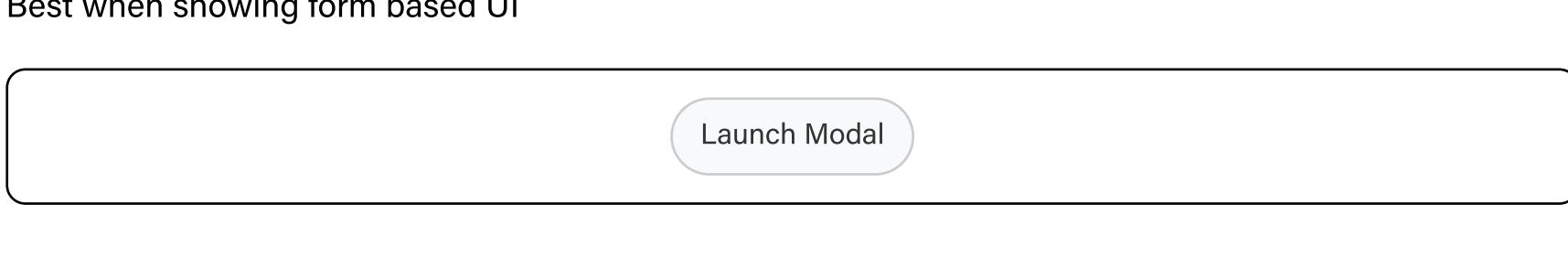
Props

<By assigned dev>

Sizes

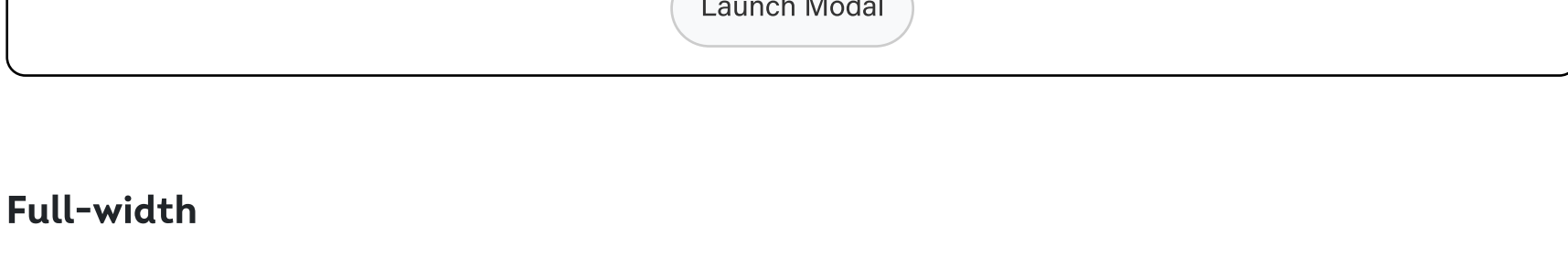
Small

Default size mainly used for short and simple tasks



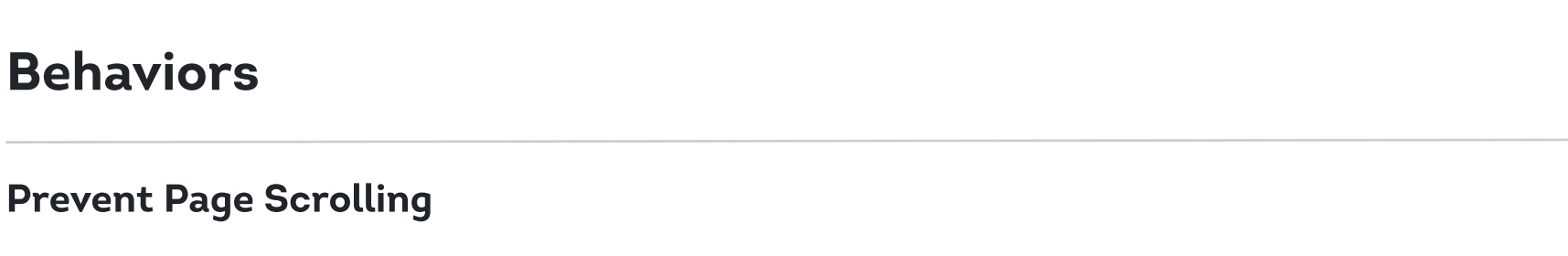
Medium

Best when showing form based UI



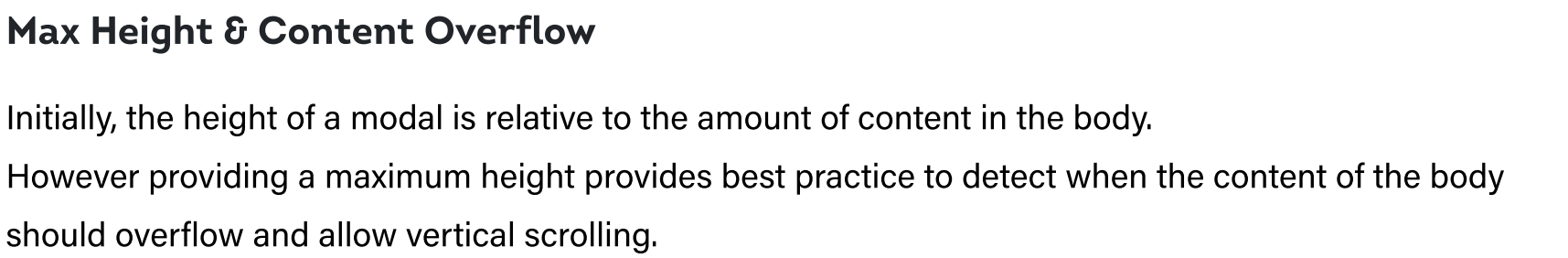
Large

Best for showing complex workflow or processes, or when horizontal space is necessary to accommodate two column layouts



Full-width

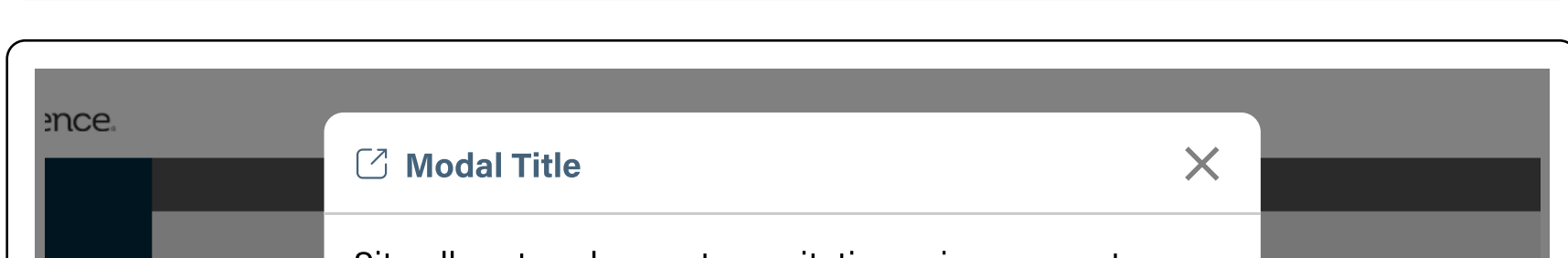
Fully customizable modal that occupies 100% of the viewport best used for free form content in different use cases



Behaviors

Prevent Page Scrolling

Once modal is opened, scrolling the main container behind the modal should be disabled

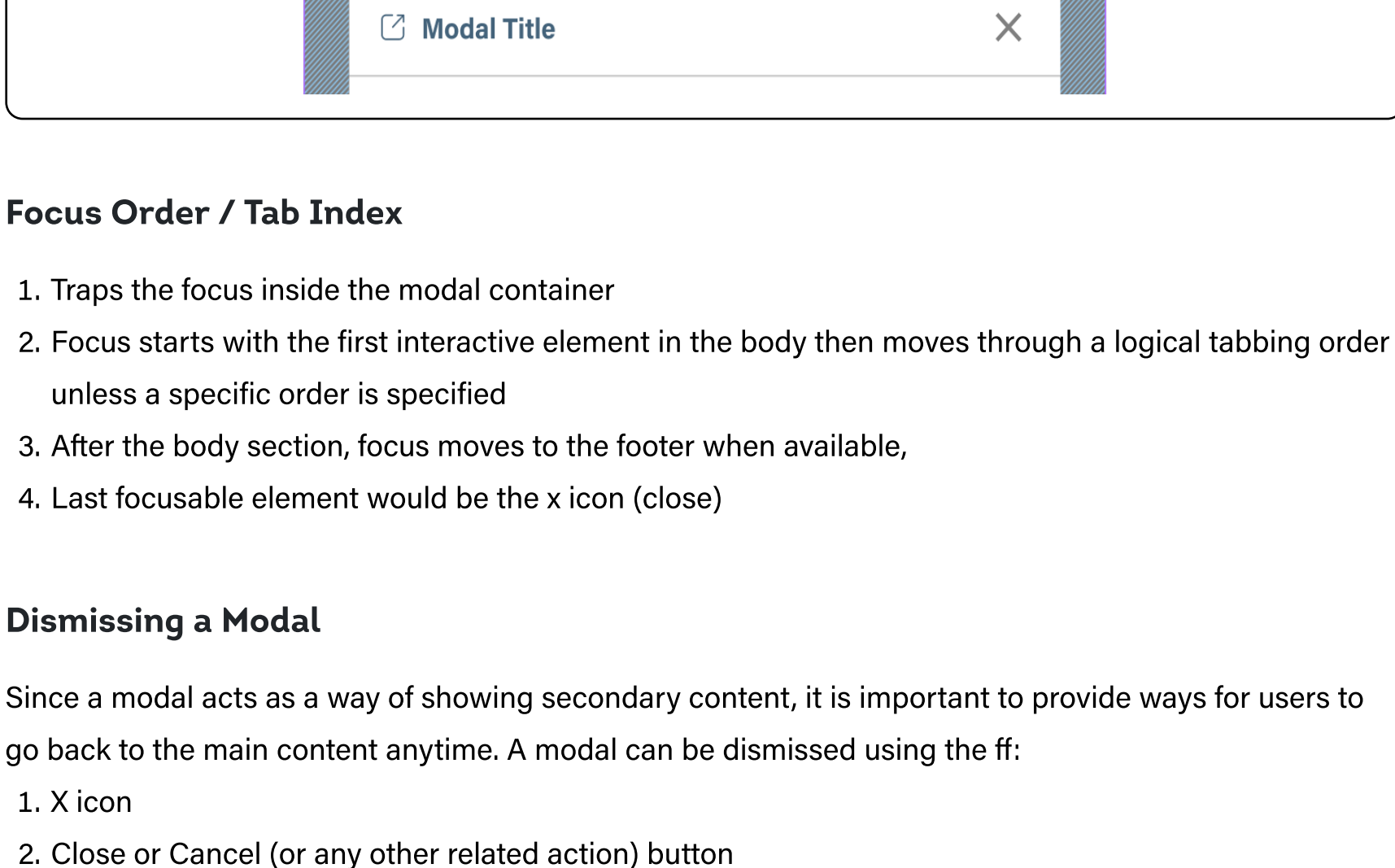


Max Height & Content Overflow

Initially, the height of a modal is relative to the amount of content in the body. However providing a maximum height provides best practice to detect when the content of the body should overflow and allow vertical scrolling.

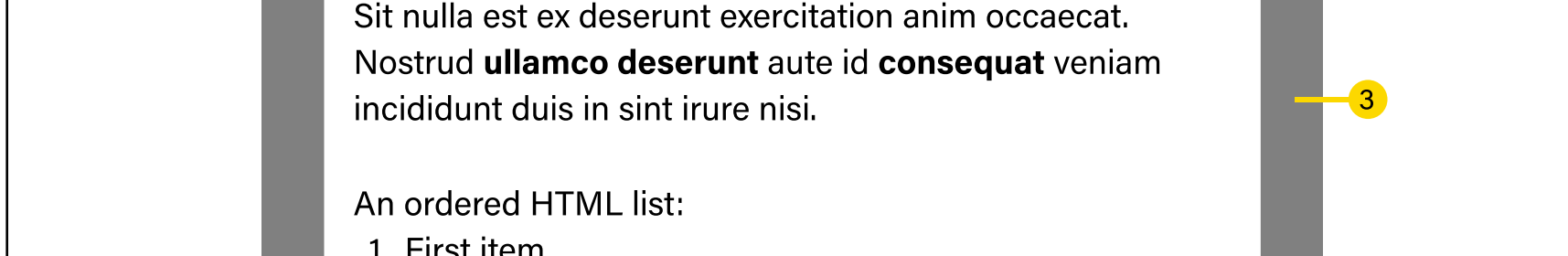
Modal max-height is determined using:
 $100vh - (top + bottom offset) = \text{modal max-height}$

Scrolling starts when body max height is:
 $100\% - (\text{header height} + \text{footer height}) = \text{body max-height}$



Position

From the top, modal has a fixed vertical offset of **spacing-5** across all sizes.



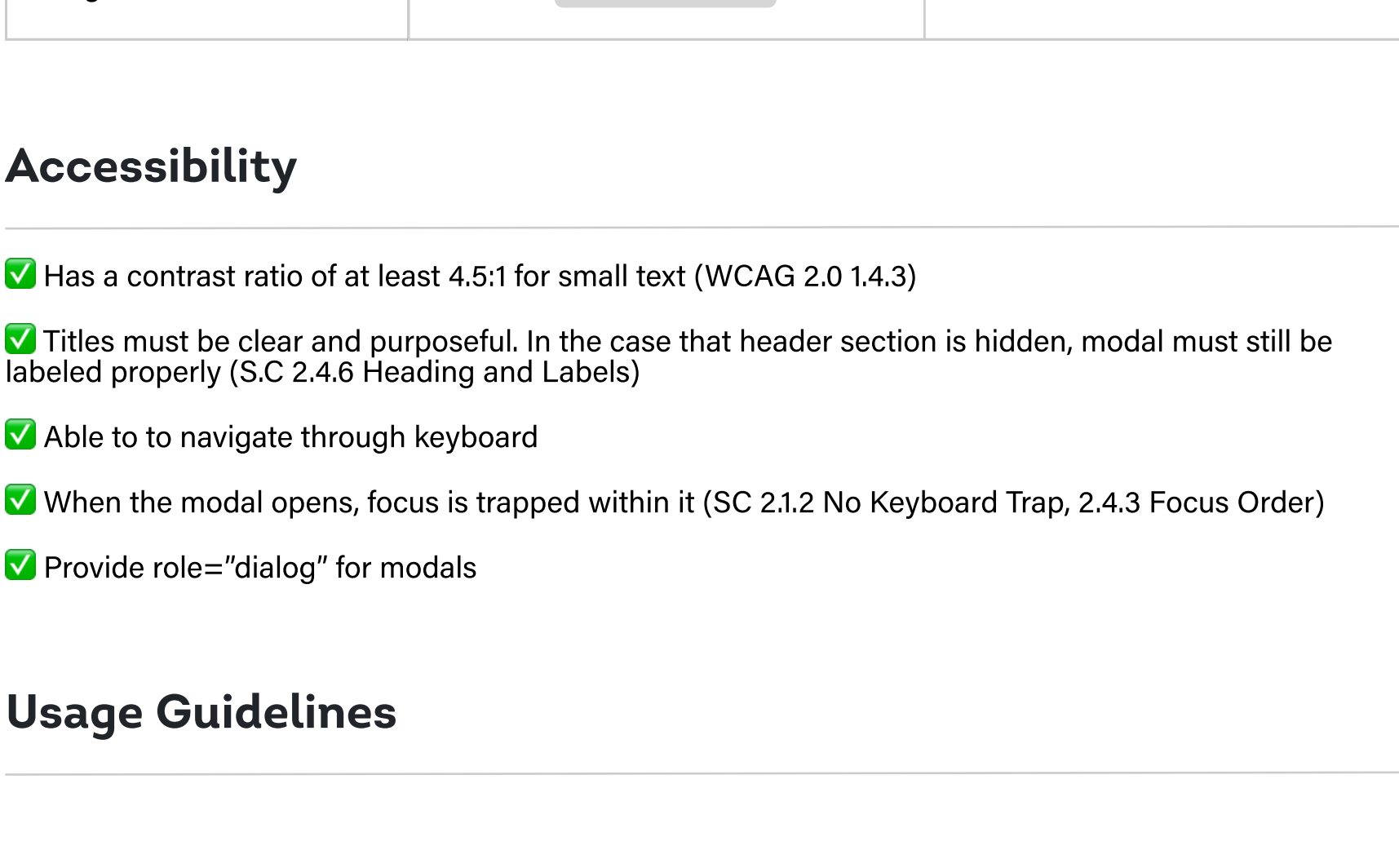
Focus Order / Tab Index

1. Traps the focus inside the modal container
2. Focus starts with the first interactive element in the body then moves through a logical tabbing order unless a specific order is specified
3. After the body section, focus moves to the footer when available,
4. Last focusable element would be the x icon (close)

Dismissing a Modal

Since a modal acts as a way of showing secondary content, it is important to provide ways for users to go back to the main content anytime. A modal can be dismissed using the ff:

1. X icon
2. Close or Cancel (or any other related action) button
3. Clicking the backdrop
4. Keyboard **Esc** key



Responsiveness

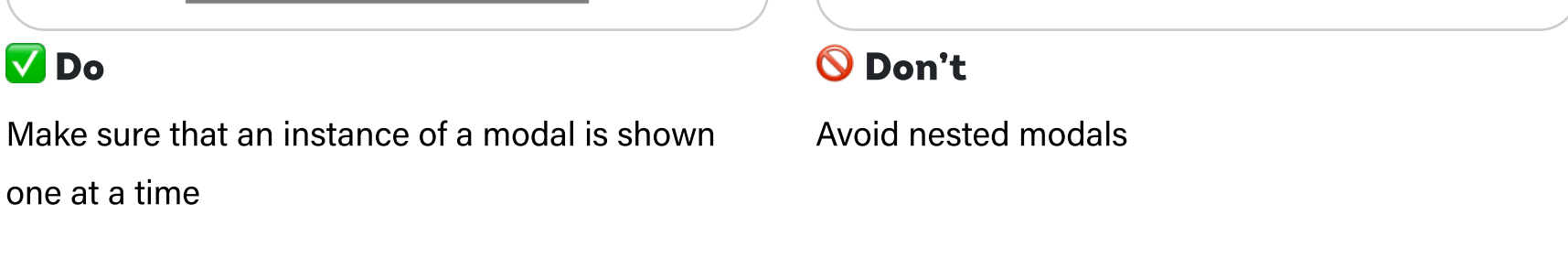
Since size is based on breakpoints, modal should stay the same size as long as breakpoint is \geq modal size.

Modal Size	Viewport	
	Other Breakpoints	Extra Small
Small	size-width-regular	100%
Medium	size-width-wide	100%
Large	size-width-full	100%

Accessibility

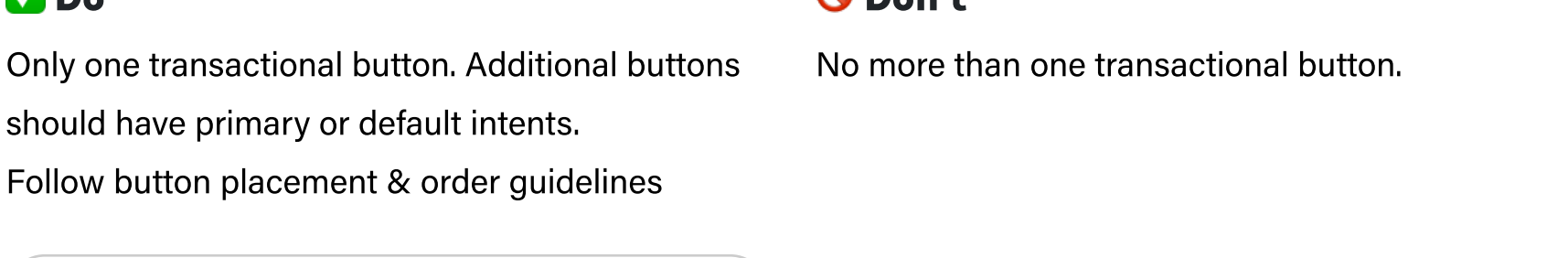
- ✔ Has a contrast ratio of at least 4.5:1 for small text (WCAG 2.0 1.4.3)
- ✔ Titles must be clear and purposeful. In the case that header section is hidden, modal must still be labeled properly (SC 2.4.6 Heading and Labels)
- ✔ Able to be navigated through keyboard
- ✔ When the modal opens, focus is trapped within it (SC 2.1.2 No Keyboard Trap, 2.4.3 Focus Order)
- ✔ Provide role="dialog" for modals

Usage Guidelines



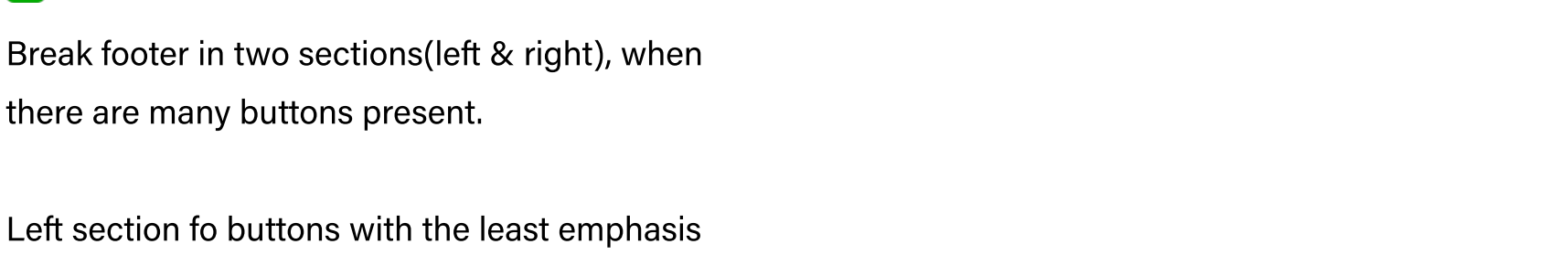
✔ **Do** Use Title case for modal headers. Keep it short and clear.

✘ **Don't** Don't use Sentence case or All-caps



✔ **Do** Use appropriate modal sizes based on intended content

✘ **Don't** Avoid making horizontally, use bigger modal sizes in this case



✔ **Do** Use modals for simple, short term tasks or processes

✘ **Don't** Avoid complex processes inside a modal



✔ **Do** Make sure that an instance of a modal is shown one at a time

✘ **Don't** Avoid nested modals

✔ **Do** Only one transactional button. Additional buttons should have primary or default intents. Follow button placement & order guidelines

✘ **Don't** No more than one transactional button.

✔ **Do** Break footer in two sections(left & right), when there are many buttons present.

Left section for buttons with the least emphasis (acknowledge, remind later,...)
Right section is for executing/triggering an event

Changelogs:

- 09/06/23 - Initial doc release (Raul)