Mayo GBS Style Guide

Prepared by the Visual Analytics Group

NCSA, University of Illinois

Sept 7, 2022

Table of Contents

Overview

Document Structure

This style guide is organized as follows:

1. **Overview -** An introduction for what is included in and how to use this document.
2. **Foundations -** Best practices recommendations for core style elements such as color, typography, layout etc.
3. **Components -** A list of components, the variations we recommend, and when/how to use them. All components in this section are selected from the PrimeNG or Ultima library and are presented in their original form (e.g., no modifications or customizations).
4. **Custom Components and Blocks -** A limited number of “blocks” customized for GBS applications. In these cases, the original PrimeNG/Ultima library components were insufficient for the type of data used in GBS applications.

Theme File

PrimeNG Stylesheet

After some investigation into the differences between PrimeNG and the Ultima “starter app” we’ve decided to create a slightly custom stylesheet that developers can use when starting their development using PrimeNG. This decision was based on conversations around the usability of Ultima and our desire to reference Mayo branding as part of this style guide. To keep it simple, however, we have done our best to adhere to styles used in the Ultima template. This will allow developers to either start with the Ultima template or start from scratch using this custom CSS file.

Developers may access this stylesheet here: https://drive.google.com/drive/folders/1TWX5-C12bQyhn-pDr0Q6Zc45sKRyowNA?usp=sharing

How to Use This Document

Navigation

Open the navigation panel in Google Docs (**View > Show Outline**). Once opened, use the clickable outline on the left panel to view the overall structure of the document, and click to navigate through this document.

Alternatively, we have included a clickable Table of Contents (pages 2-7).

Contact

If you have any questions or suggestions for improvement, please contact us:

- **Email:** fangyu@illinois.edu

- **Leave a comment on Google Doc:** Add a comment on this Google doc; tag all comments with @fangyu@illinois.edu

- **Leave a comment on Figma:** You can find the link to Figma files in the 'Custom Components and Blocks' section.

  How to use the comment function in Figma:

  1.) Hit 'C' - Alternatively, click the 'thought bubble' icon on the top bar.
  2.) Click on the location related to your questions and add a comment.

Foundations

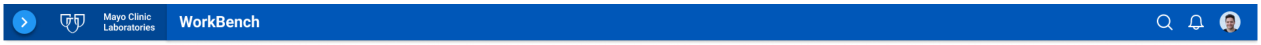
Within this section, you will find common styles, assets and guidance which will apply no matter what application context you are working with. Some of this guidance will be automatically implemented by the PrimeNG framework for some pre-built components, however, it is good to acquaint yourself with best practices in the event that you are required to design custom components.

Generally, these are recommendations for common user interface and interaction patterns. Use these to help guide you in your own development and design decisions for building custom components and blocks.

Mayo Branding

Use of Mayo Logo

In order to make using the Mayo brand as consistent as possible with the Mayo Brand Guidelines as well as across multiple, related applications, we are recommending using the **white-only version** of the Mayo Shield logo as illustrated here on the “topbar” component:



Mayo Blue

Mayo Blue Color	#0057B8	🟦

Application Name Styling

Recommendations for styling of the app name/identity text:

Text Color	#FFFFFF
Type Size	H3 or 3XL
Type Weight	Class: `font-bold` Properties: `font-weight: 700`

Refer to the image above for placement guidance or view the figma mockup.

Top Bar Styling

We are deviating from the style choices provided by the Ultima theme/template and recommending that **all applications use the "Mayo blue" color (#0057B8) for the top bar.**

This will be included in the CSS file provided with this documentation.

Color

Judicious use of color makes the visual design more consistent, provides meaning & adds hierarchy to the interface.

General Colors

To coordinate with the Mayo brand blue used for the top bar we recommend using the following as the primary color for the application. This will be used for buttons, and to show active or selected states for components.

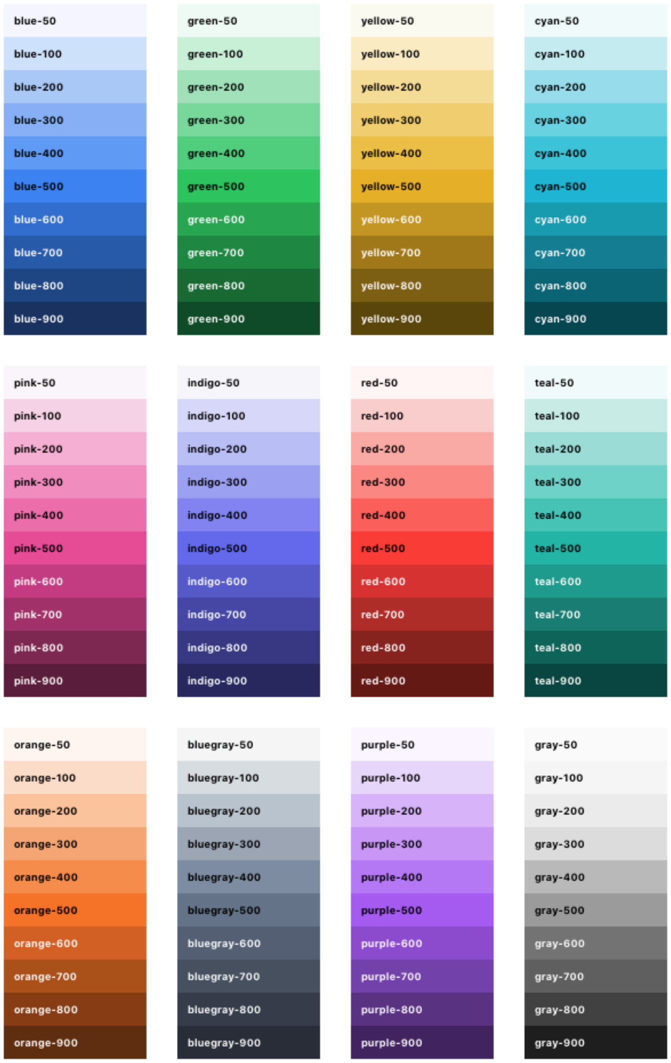
Variable	Value	Visual	Description
`--text-color`	rgba (0,0,0 0.87)	■	Font text color.
`--text-secondary-color`	rgba (0,0,0 0.6)	■	Muted font text color with a secondary level.

--primary-color	#2196F3	🟦	Primary color of the theme.
—primary-lightest-color	#E3F2FD	⬜	Light color based on Primary that can be used for defining backgrounds of components in certain states as well as functional areas of the application.
--primary-color-text	#FFFFFF	⬜	Text color when the background is the primary color.
-errorColor	#B00020	🟥	

Color Palettes

Within the PrimeNG framework, a palette consists of 10 colors where each color provides tints/shades from 50 to 900. We recommend the blue color palette from the Ultima stylesheet:

PrimeNG Palettes



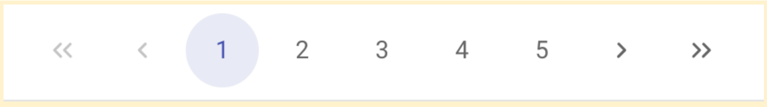
Primary Color

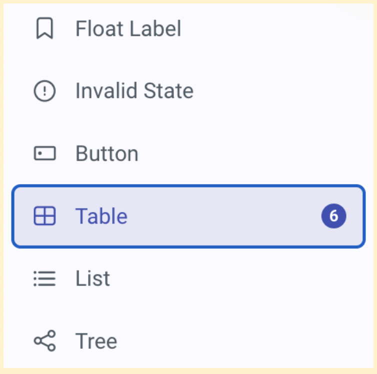
Below is the palette containing the primary color we are recommending:

Blue 50	f4fafe	
Blue 100	cae6fc	
Blue 200	a0d2fa	
Blue 300	75bef8	
Blue 400	4baaf5	
Blue 500	**2196f3**	**Primary Color**
Blue 600	1c80cf	
Blue 700	1769aa	
Blue 800	125386	
Blue 900	0d3c61	

Tints and Shades

For many base components, PrimeNG automatically uses tints and shades of the selected Primary color in certain cases such as hover states and selected states for some components. Some examples include:





Tints (primary color + white) reference the primary color but place the component further back in the hierarchy of the application. When applying tints as part of a component life cycle, consider how they have been used in pre-built PrimeNG components. If the tint is used to indicate state, consider using the same tint that is employed by the framework.

Ultima/PrimeNG Tints and Shades

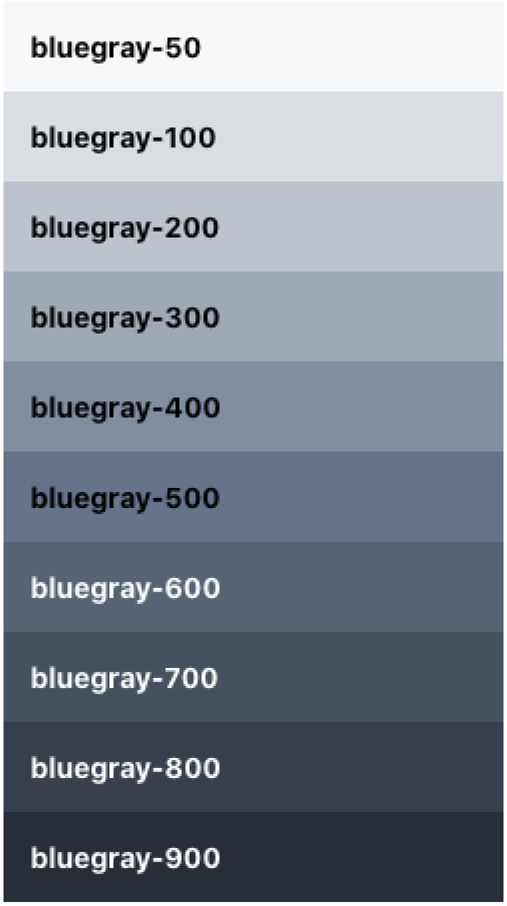
Variable	Value	Visual	Description
`--surface-hover`	#000000 4%	⬜	Value often used in the framework for hover states for components.
`--buttonBG-4%`	#2196f3 4%	🟦	Color percentage value of a design token such as primary color (shown here) or success, warning, danger, that is used in the Ultima/PrimeNG framework. Consider using this percentage value when you require a subtle color to indicate state or to define an area of the application or component such as a toolbar, header, or content area.
`--buttonBG-16%`	#2196F3 16%	🟦	Color percentage value of a design token such as primary color (shown here) or success, warning, danger,etc that is used in the Ultima/PrimeNG framework. The framework uses this color value to indicate focus in several of it’s components.
`-highlightBG`	#E3F2FD	🟦	*Note: Same color as primary-lightest-color

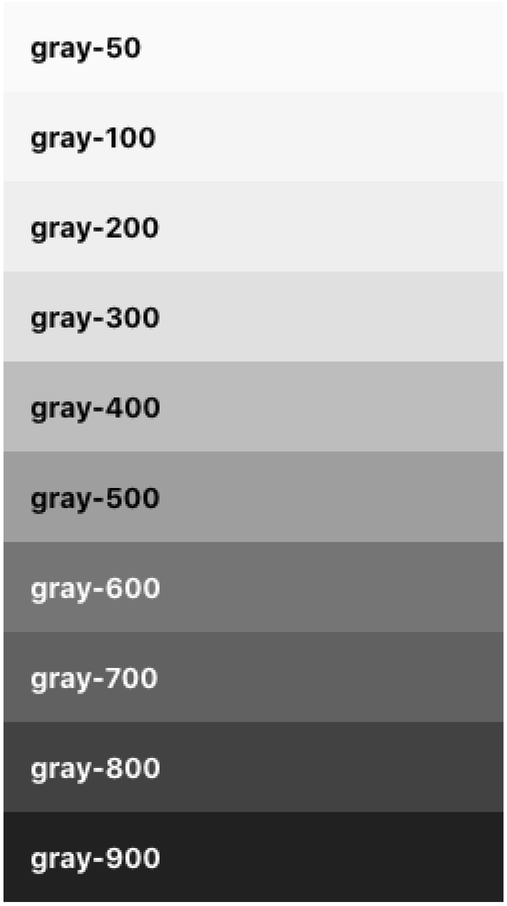
Neutrals

A neutral color palette articulates the boundaries of components and interface. PrimeNG and Ultima have cooler and warmer neutral color palettes.

The Ultima theme uses the Gray palette for some of its components. Some of the following palette is represented in the “Surfaces” palette.

PrimeNG Neutrals





* Palette used by the framework

Surfaces

The Surfaces palette articulates the canvases on which interfaces and components are deployed. Ultima commonly uses the following tags to define surfaces.

Commonly Used Tags

Variable	Value	Visual	Description
`--surface-ground`	#FAFAFA		Color on which all components rest.
`--surface-section`	#FFFFFF		

--surface-card	#FFFFFF		Color of components that use the “Card” component as their base component.
—surface-overlay	#FFFFFF		
--surface-border	rgba(0,0,0,.12)	⬜	
--surface-hover	rgba(0,0,0,.04)	⬜	

Status Colors

There are 3 general statuses: Success, Warning and Danger included within the framework. Each status has a color that can be used in icons, borders and texts.

Success

Variable	Value	Visual	Description
--successBG	#689F38	🟩	The Success color is used in positive conditions, for example in high scores, accomplished tasks, etc.
--successButtonHoverBg -92%	#689F38 92%	🟩	Variant of Success color. Hover background state for Success button. Very slight color shift!
--buttonBG-68%	#689F38 68%	🟩	Variant of Success color. Active background state for Success button.
Success message text color	#1B5E20	🟩	Success text when used on light background
success message background color	#C8E6C9	🟩	Success light background color

Warning

Variable	Value	Visual	Description
--warningBG	#FBC02D	🟨	The Warning color is used to warn the user about something important but without the seriousness of an error. It can also be used to illustrate the concept of “middle” or “medium” as in status.
--warningButtonHoverBg -92%	#689F38 92%	🟨	Variant of Warning color. Hover background state for Warning button. Very slight color shift!

--warningBG-68%	#689F38 68%	🟨	Variant of Warning color. Active background state for Warning button.
warning message text color	#7f6003	🟫	Warning text when used on light background
warning message background color	#FFECB3	🟨	Warning light background color

Danger

Variable	Value	Visual	Description
--dangerBG	#D32F2F	🟥	The Danger color is used to warn the user about something important but without the seriousness of an error. It can also be used to illustrate the concept of “middle” or “medium” as in status.
--dangergButtonHoverBg -92%	#689F38 92%	🟥	Variant of Danger color. Hover background state for Danger button. Very slight color shift!
--dangerBG-68%	#689F38 68%	🟥	Variant of Danger color. Active background state for Danger button.
danger message text color	#B71C1C	🟥	Danger text when used on light background
danger message background color	#FFCDD2	🟥	Danger light background color

Other Status Colors

GBS applications often use color to indicate qualitative information about various statuses/states beyond the above very generic categories. In these cases our recommendations would be:

- Choose from the PrimeNG palettes for consistency across applications
- Limit the number of colors used as much as possible
- Consider using lighter shades of a color to reduce the amount of visual noise in the application and support the hierarchy and primary workflow of the interface.

Usage Tips

- **Focus Attention** - The use of color should help bring attention to what matters most. Color should support the hierarchy of the page.
- **Minimal Palette** - Refrain from using palettes other than the primary palette. Introduction of other colors creates confusion and obscures the hierarchy of the application.
- **Consistency** - Do not use the variants of the primary color indiscriminately. If you choose to use a variant of primary, try to use it for the same/similar function across the application.
- **Accessibility** - Consider color blindness and contrast when selecting colors.

Typography

Correct use of typography is one of the primary ways to establish hierarchy in an interface. For many low-level components in the Prime NG library, these typographic decisions have already been made for you. However, when building custom blocks from PrimeNG components, please consider the following recommendations for when/where to use various type styles.

Overview

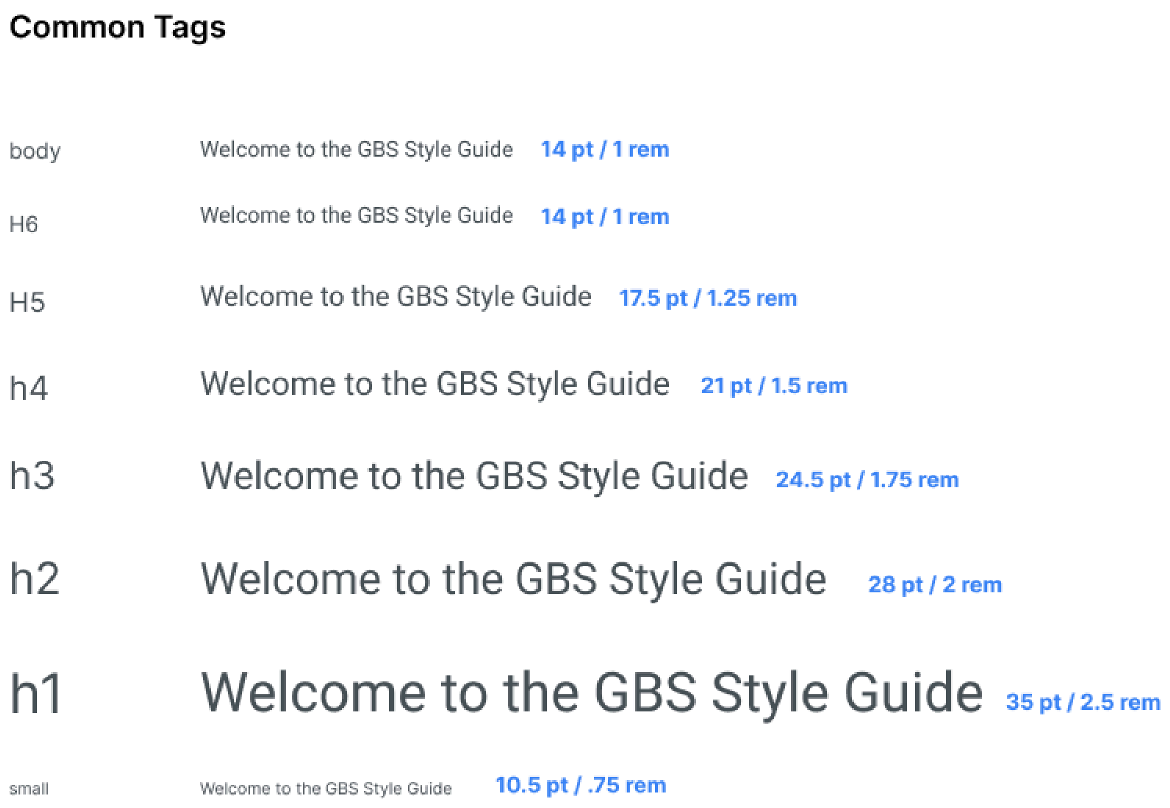
The primary font family in the Ultima theme is **Roboto**.

```
$fontFamily:Roboto,Helvetica Neue Light,Helvetica Neue,Helvetica,Arial,Lucida Grande,sans-serif;
```

Roboto is a Google font available for free here:
https://fonts.google.com/?query=roboto

Font Sizes

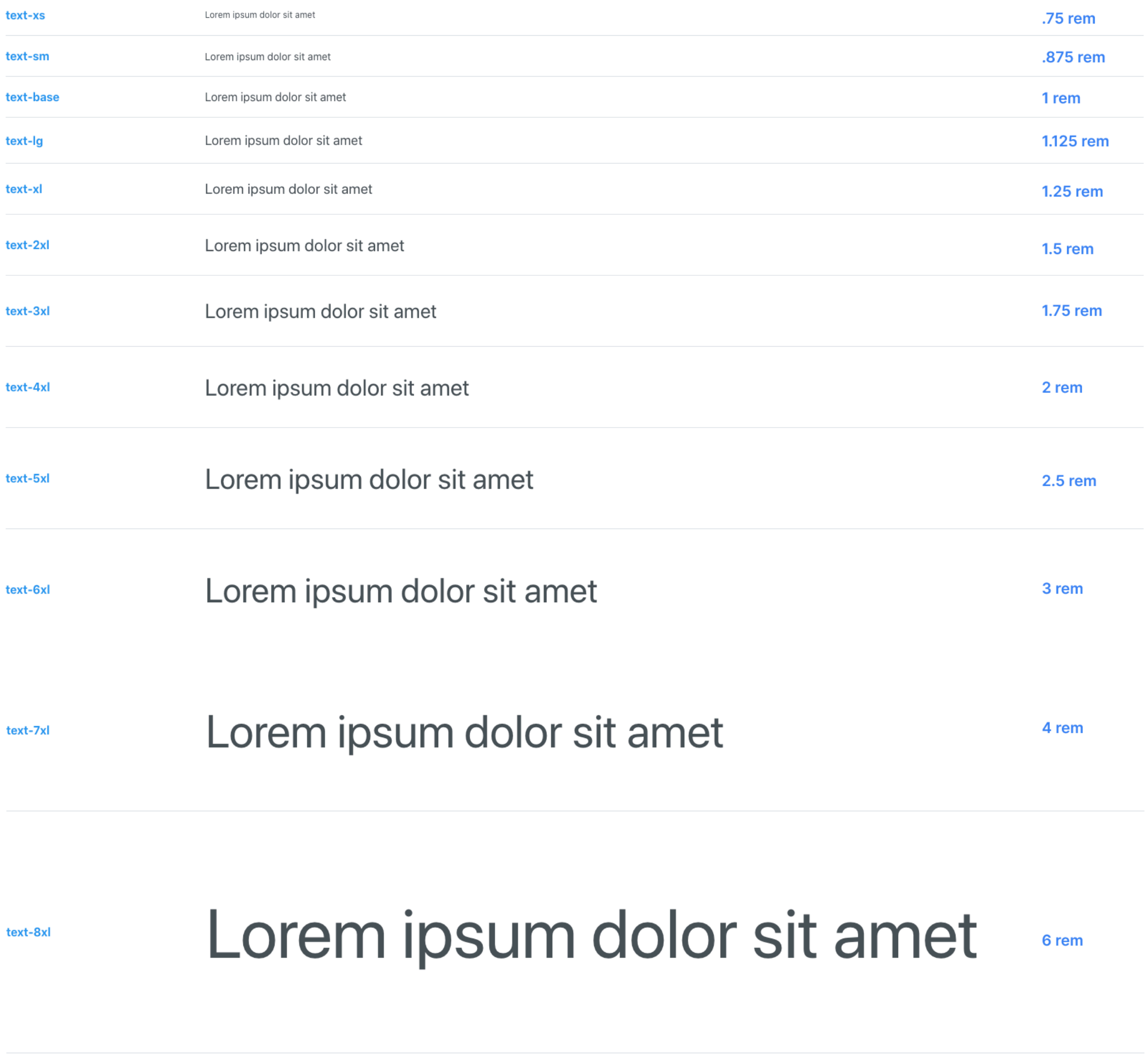
PrimeNG provides the following type sizes as part of the stylesheet definition. Note: rem sizes are calculated from the base font size. In the case of the Ultima theme/template, **Roboto 14pt = 1 rem.**



Common Tags

body	Welcome to the GBS Style Guide	14 pt / 1 rem
H6	Welcome to the GBS Style Guide	14 pt / 1 rem
H5	Welcome to the GBS Style Guide	17.5 pt / 1.25 rem
h4	Welcome to the GBS Style Guide	21 pt / 1.5 rem
h3	Welcome to the GBS Style Guide	24.5 pt / 1.75 rem
h2	Welcome to the GBS Style Guide	28 pt / 2 rem
h1	Welcome to the GBS Style Guide	35 pt / 2.5 rem
small	Welcome to the GBS Style Guide	10.5 pt / .75 rem

For added flexibility, PrimeNG also uses font scaling from XS to 8XL as shown below. There is some overlap between the html tags and the PrimeNG tags for type scaling so these are sometimes used interchangeably in the stylesheet..

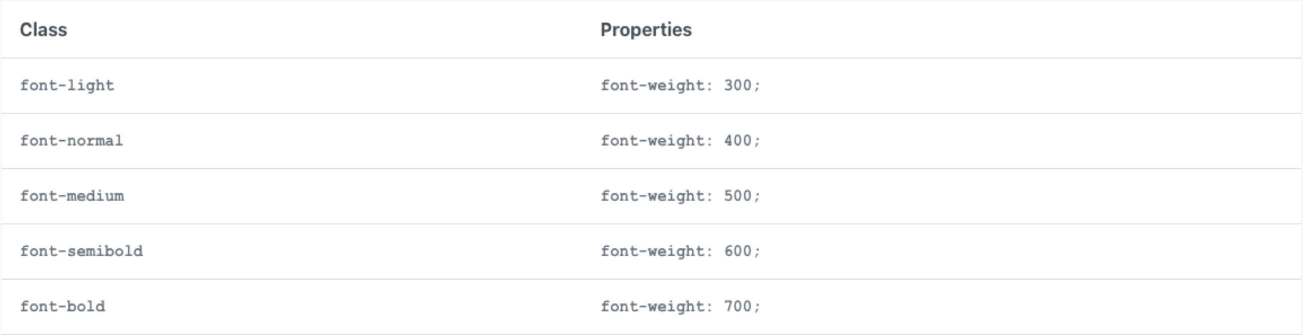


text-xs	Lorem ipsum dolor sit amet	.75 rem
text-sm	Lorem ipsum dolor sit amet	.875 rem
text-base	Lorem ipsum dolor sit amet	1 rem
text-lg	Lorem ipsum dolor sit amet	1.125 rem
text-xl	Lorem ipsum dolor sit amet	1.25 rem
text-2xl	Lorem ipsum dolor sit amet	1.5 rem
text-3xl	Lorem ipsum dolor sit amet	1.75 rem
text-4xl	Lorem ipsum dolor sit amet	2 rem
text-5xl	Lorem ipsum dolor sit amet	2.5 rem
text-6xl	Lorem ipsum dolor sit amet	3 rem
text-7xl	Lorem ipsum dolor sit amet	4 rem
text-8xl	Lorem ipsum dolor sit amet	6 rem

PrimeNG typography: https://www.primefaces.org/primeflex/fontsize

Font Weight

PrimeNG offers a limited range of weights for the typeface:

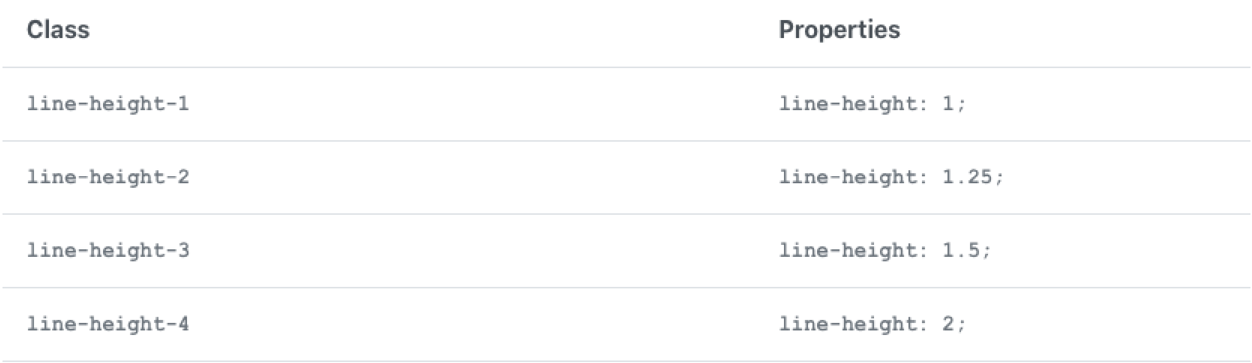


Class	Properties
`font-light`	`font-weight: 300;`
`font-normal`	`font-weight: 400;`
`font-medium`	`font-weight: 500;`
`font-semibold`	`font-weight: 600;`
`font-bold`	`font-weight: 700;`

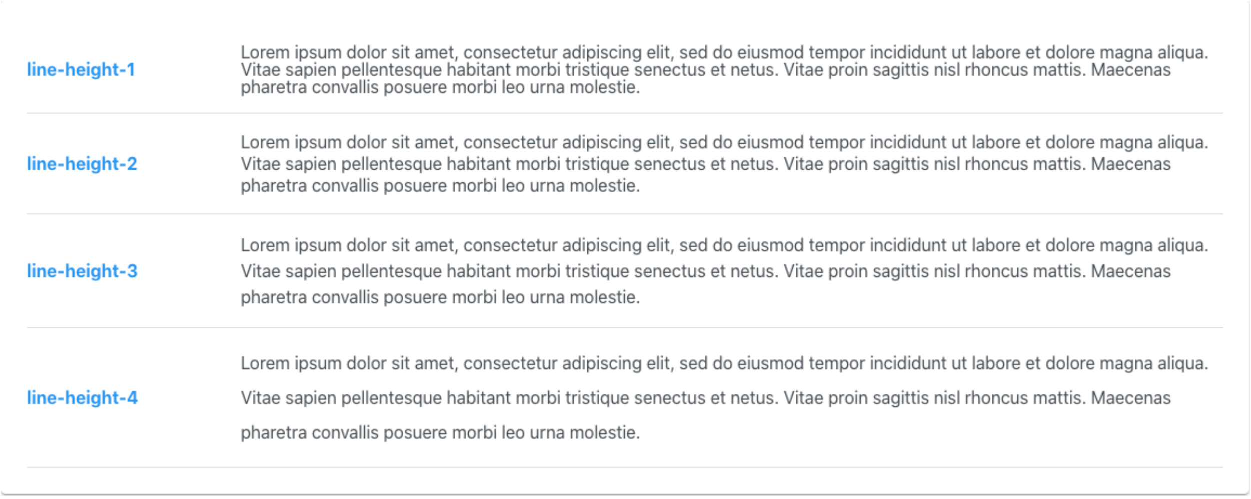
Of those, the weights **most commonly used appear to be 400, 500, and 600**.

Line Height

PrimeNG provides four classes for Line Height:



Class	Properties
`line-height-1`	`line-height: 1;`
`line-height-2`	`line-height: 1.25;`
`line-height-3`	`line-height: 1.5;`
`line-height-4`	`line-height: 2;`



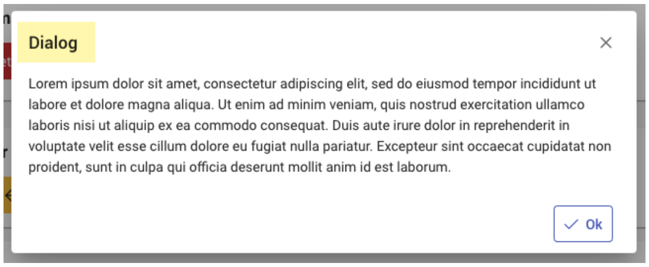
line-height-1	Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Vitae sapien pellentesque habitant morbi tristique senectus et netus. Vitae proin sagittis nisl rhoncus mattis. Maecenas pharetra convallis posuere morbi leo urna molestie.
line-height-2	Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Vitae sapien pellentesque habitant morbi tristique senectus et netus. Vitae proin sagittis nisl rhoncus mattis. Maecenas pharetra convallis posuere morbi leo urna molestie.
line-height-3	Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Vitae sapien pellentesque habitant morbi tristique senectus et netus. Vitae proin sagittis nisl rhoncus mattis. Maecenas pharetra convallis posuere morbi leo urna molestie.
line-height-4	Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Vitae sapien pellentesque habitant morbi tristique senectus et netus. Vitae proin sagittis nisl rhoncus mattis. Maecenas pharetra convallis posuere morbi leo urna molestie.

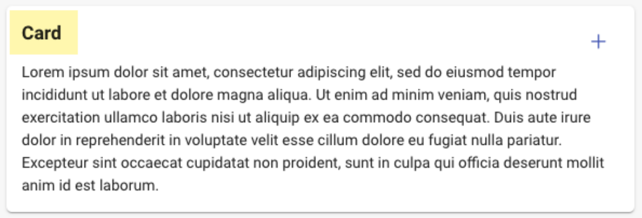
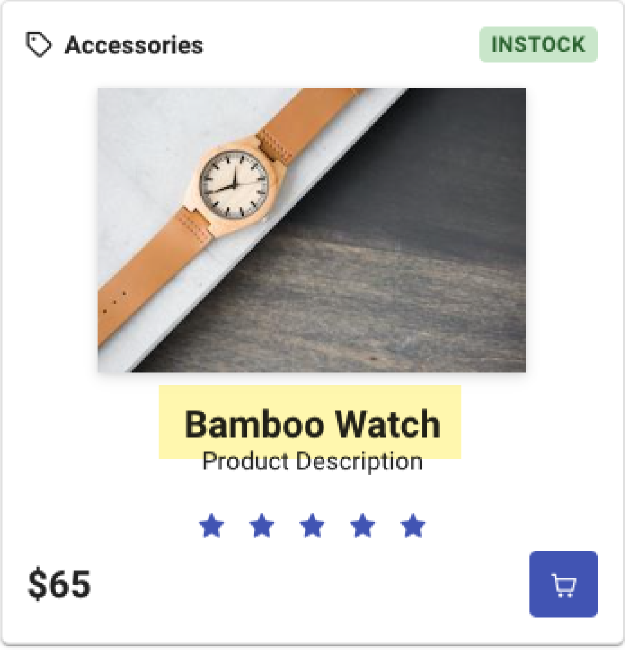
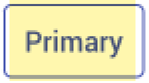
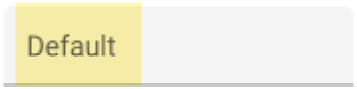
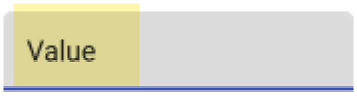
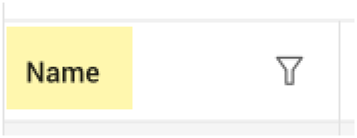
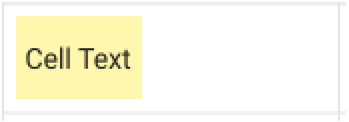
These classes are already embedded in the pre-defined type styles.

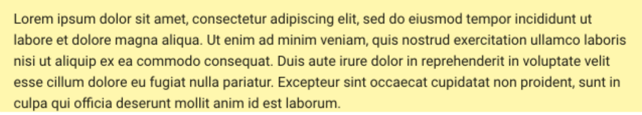
We do not recommend using `line-height-1` as the lack of spacing makes for difficult reading. `line-height-2` and `line-height-3` are more suited to GBS application needs. If the use case allows, a little more space is easier on the eyes and aids reading comprehension.

Common Styles

Many of the typography style choices have been made for you within the PrimeNG framework. Following is a list of styles commonly used and their UI contexts. Use these as a guide when building custom components where you need to make typographic choices.

Component	Size	Weight	
Dialogue Title	H5 or text-xl	500, medium	 Dialog Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Card Title	H5 or text-xl	600, semi-bold	 Card + Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.
Alt Card Title	H4 or text-2xl	700, bold	 Accessories INSTOCK Bamboo Watch Product Description ★ ★ ★ ★ ★ $65
Button Label	H6 or text-base	500, medium	 Primary
Input Field Label	H6 or text-base	400, normal	 Default
Input Field Value	H6 or text-base	400, normal	 Value
Table Column Label	H6 or text-base	500, medium	 Name
Table Cell Text	H6 or text-base	400, normal	 Cell Text

Paragraph Text	H6 or text-base	400, normal	 Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Usage Tips

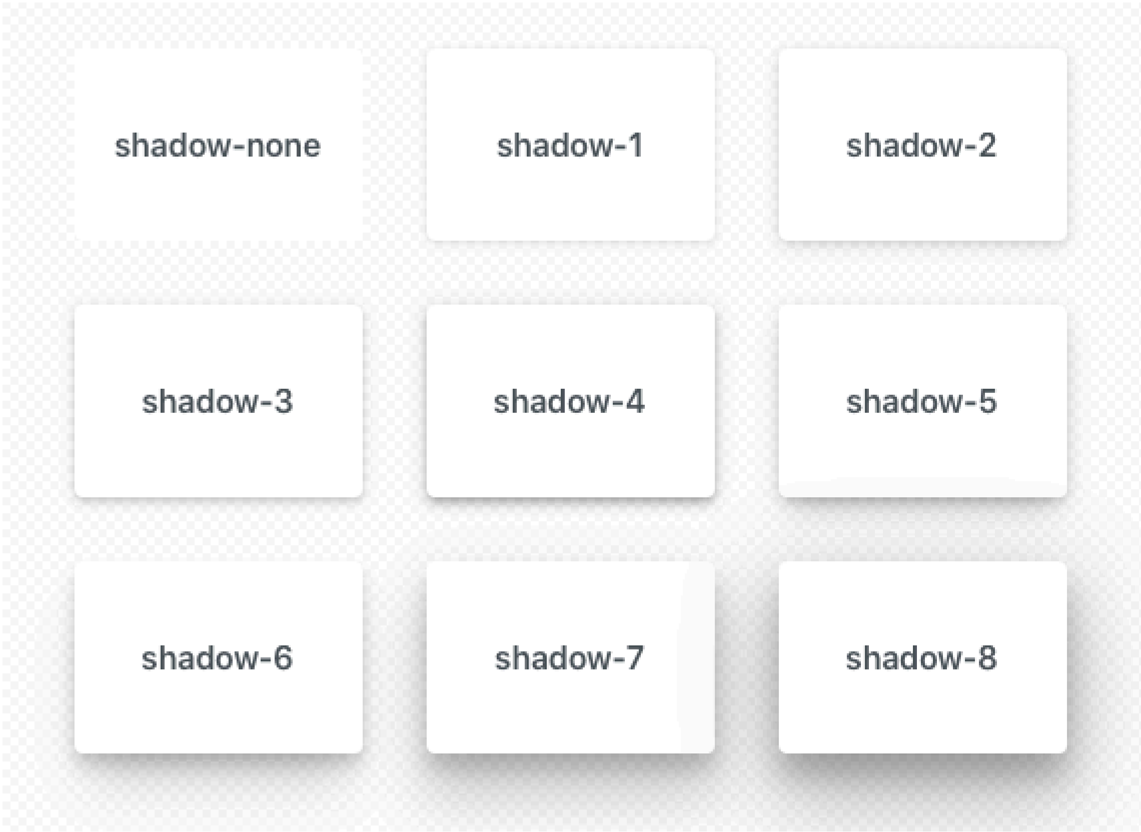
- **Keep the number of text styles used within an interface to a minimum and be consistent across the application with the styles you choose.**
- **Don't let paragraph widths get too thin or too long** - Paragraphs of text that are too long are difficult to follow, and paragraphs of text that are too thin are difficult to read. Ideally, blocks of text should be roughly 70 characters wide. Be sure to keep them at least between 50 and 120 characters wide.
- **Keep content short and to the point** - Keep paragraphs, tooltips, descriptions concise. Especially with data-intensive applications, brevity is important for the overall readability and pleasant user experience.
- **Consider scannability -** when choosing styles, choose lighter weights and shades for labels and heavier weights/shades for associated values.
- **Follow the framework** - When building custom blocks, base your typographic decisions on components that are similar in function or have a similar place in the user journey.

Elevation

The surface on which content and components live. It is a visual motif used to create hierarchy and focus through depth.

PrimeNG/Ultima Styles

There are many elevations available within the PrimeNG framework and decisions about which to use for which components have been made by the framework. When designing custom components, consult the standards that have been set by the framework. Elevation style presets are as follows:



https://www.primefaces.org/primeflex/elevation

The Ultima theme/template seems to use custom styles for elevations it uses for some of its components. After a visual review it seems that these custom elevation styles are most similar to **shadow-1** (for card styles) and **shadow-8** (for overlays).

Usage Tips

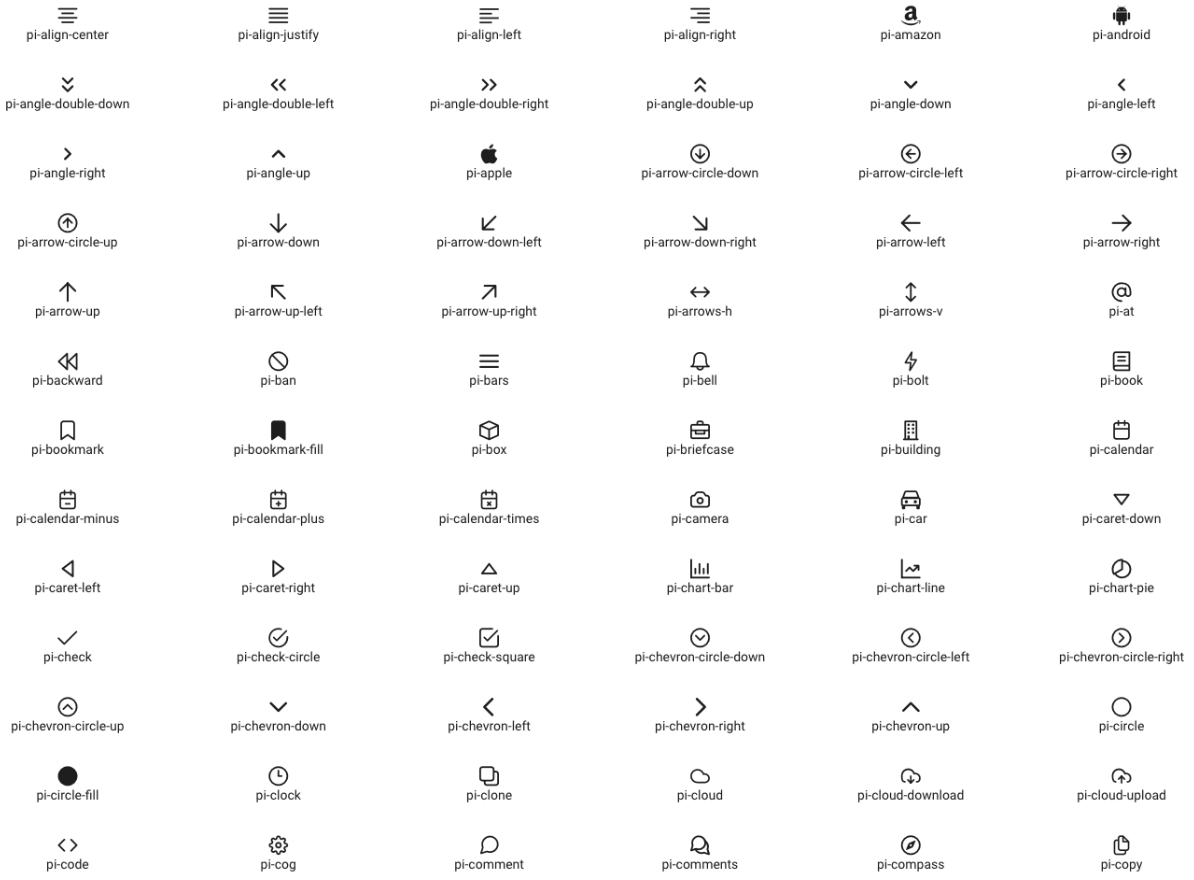
- Use elevations sparingly and follow the framework for guidance on where/when to apply elevation styles.
- Study the Ultima stylesheet for custom elevation configurations and follow those in similar use cases.

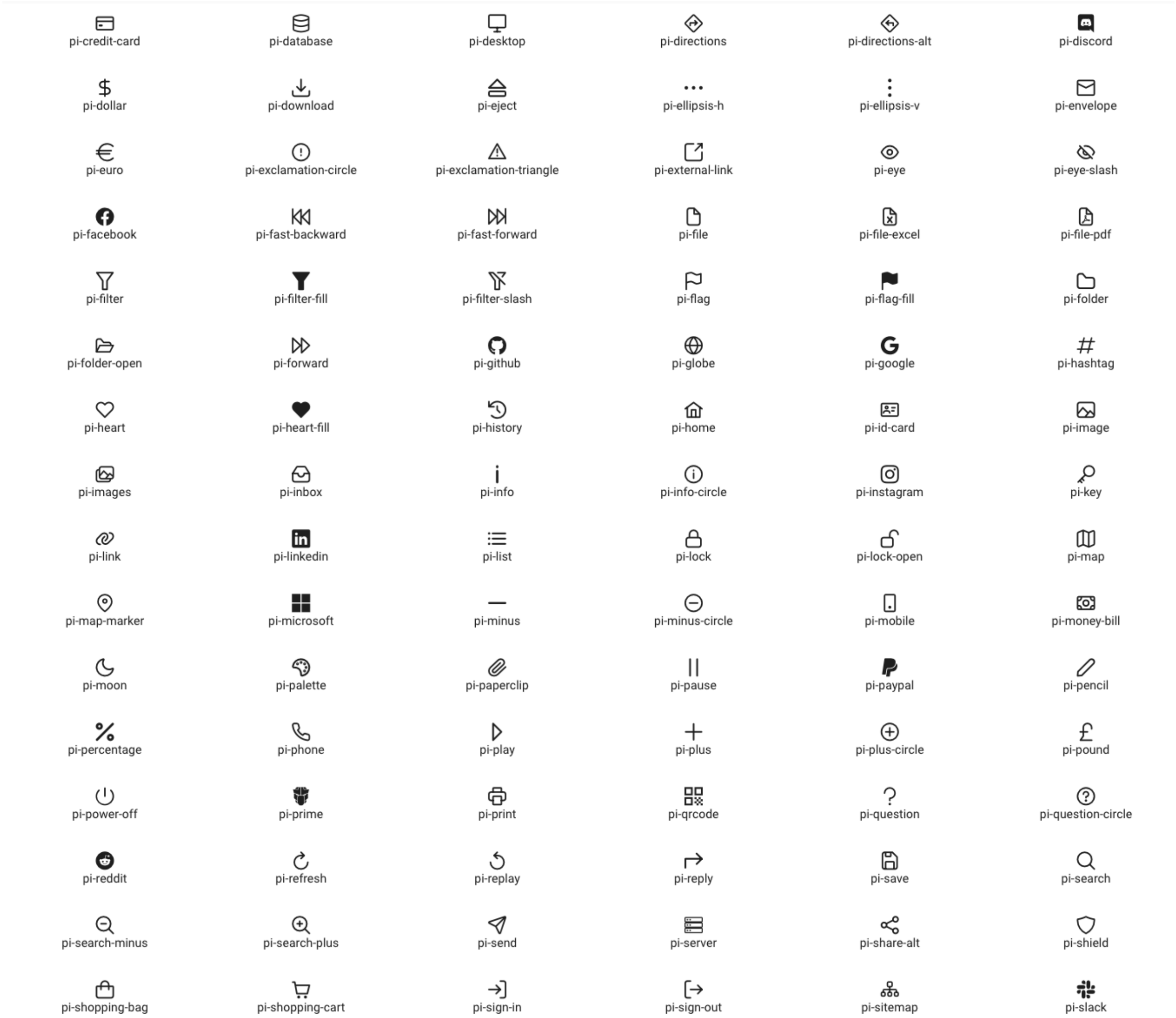
Icons

Icons support various actions and metaphors. They should be simplified graphic forms so that they work at small sizes and combine nicely with the typographical context. Varying styles across icon sets should be avoided as this creates a feeling of disharmony within the application.

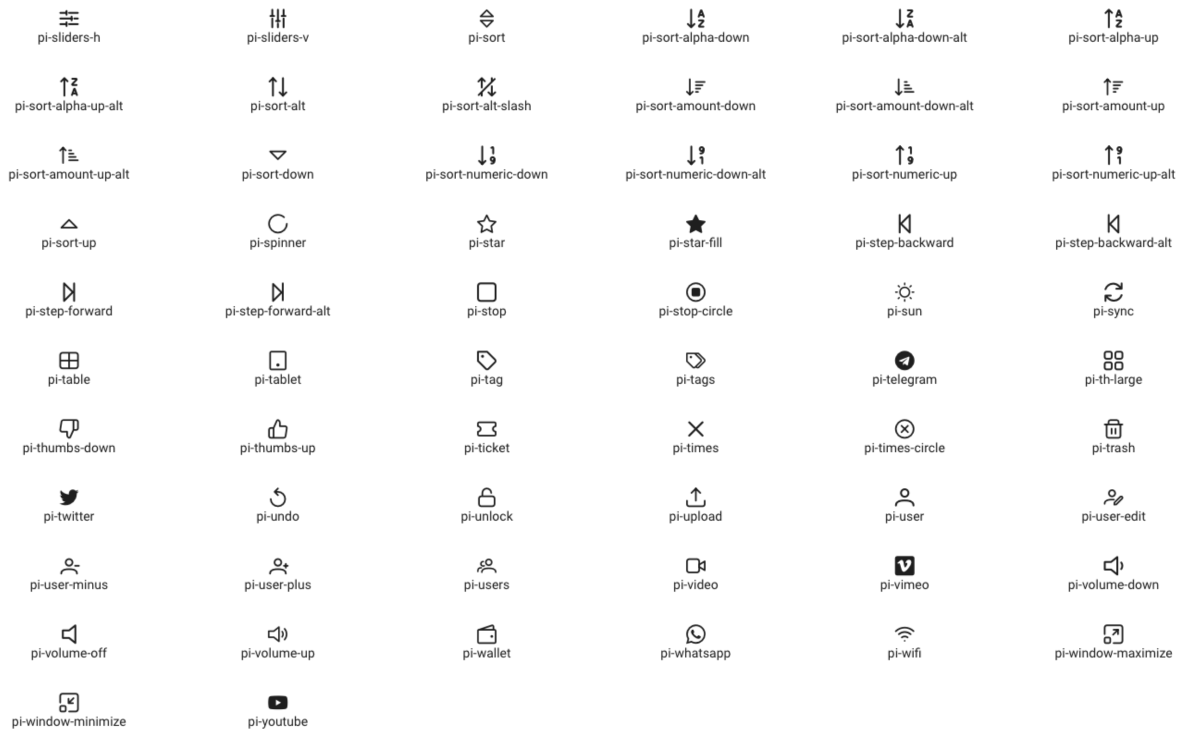
Prime Icons

Prime offers a fairly comprehensive set of icons for common usages.





pi-credit-card	pi-database	pi-desktop	pi-directions	pi-directions-alt	pi-discord
pi-dollar	pi-download	pi-eject	pi-ellipsis-h	pi-ellipsis-v	pi-envelope
pi-euro	pi-exclamation-circle	pi-exclamation-triangle	pi-external-link	pi-eye	pi-eye-slash
pi-facebook	pi-fast-backward	pi-fast-forward	pi-file	pi-file-excel	pi-file-pdf
pi-filter	pi-filter-fill	pi-filter-slash	pi-flag	pi-flag-fill	pi-folder
pi-folder-open	pi-forward	pi-github	pi-globe	pi-google	pi-hashtag
pi-heart	pi-heart-fill	pi-history	pi-home	pi-id-card	pi-image
pi-images	pi-inbox	pi-info	pi-info-circle	pi-instagram	pi-key
pi-link	pi-linkedin	pi-list	pi-lock	pi-lock-open	pi-map
pi-map-marker	pi-microsoft	pi-minus	pi-minus-circle	pi-mobile	pi-money-bill
pi-moon	pi-palette	pi-paperclip	pi-pause	pi-paypal	pi-pencil
pi-percentage	pi-phone	pi-play	pi-plus	pi-plus-circle	pi-pound
pi-power-off	pi-prime	pi-print	pi-qrcode	pi-question	pi-question-circle
pi-reddit	pi-refresh	pi-replay	pi-reply	pi-save	pi-search
pi-search-minus	pi-search-plus	pi-send	pi-server	pi-share-alt	pi-shield
pi-shopping-bag	pi-shopping-cart	pi-sign-in	pi-sign-out	pi-sitemap	pi-slack



Ultima: https://www.primefaces.org/ultima-ng/#/utilities/icons

PrimeFaces: http://primefaces.org/primeng/icons

Note: The above set is, primarily, outline style, which is associated with slower comprehension and does not scale down as well as filled versions. You may consider replacing this set with a filled version in the future.

Custom Icons

As a result of our review of GBS apps, we have designed a few custom icons that do not have corollaries in the Prime library. These can be found here:

View the Figma file

Download the SVG files of customized icons

Please contact fangyu@illinois.edu if you need further customized icons.

Color

PrimeNG/Ultima applies neutral colors to icons in many example components. A common theme color used for icons is:

`--text-secondary-color`	rgba (0,0,0 0.6)	■	Muted font text color with a secondary level. Also used for icons.

Size

Within the framework, icon sizes are designated in rem units. Size of icons can be changed by using **`font-size`** property.

Common sizes are 1 rem and 1.71 rem. Refer to application of icons in existing components to guide your sizing of icons

Usage Tips

Avoid Overuse - Icons offer powerful visual support but overuse clutters the interface and defeats their purpose which is to provide visual cues and improve scannability.

Consider Hierarchies - First, work on general layout, then look for labels that need an extra visual cue. Consider hierarchies to determine where/when icons should be added.

Be Consistent - Be sure to use the same color and size for icons that are used in similar ways across your application. This reinforces a sense of order, logic and the overall hierarchy of the application.

Usage Options

Icon only

- For well-established metaphors. (Trash for deleting, Pencil for edit, ...)

- If the context doesn’t support a label (Action buttons in data grid rows, Close buttons in modals, …)

Icon plus label

- For highlighting a concept or action
- To support the meaning of the label

Layout & Spacing

Empty space in an interface reinforces hierarchies and allows the user to distinguish related content and elements on the screen.

PrimeNG/Ultima Grid

PrimeFlex uses a 12 column based grid for page layout. Use this grid when developing applications from scratch (vs. using the Ultima) template.

https://www.primefaces.org/primeflex/gridsystem

PrimeNG/Ultima Spacing

Primeflex steers developers toward specifying margin and padding in increments of **0.25rem**; see https://www.primefaces.org/primeflex/margin and https://www.primefaces.org/primeflex/padding .

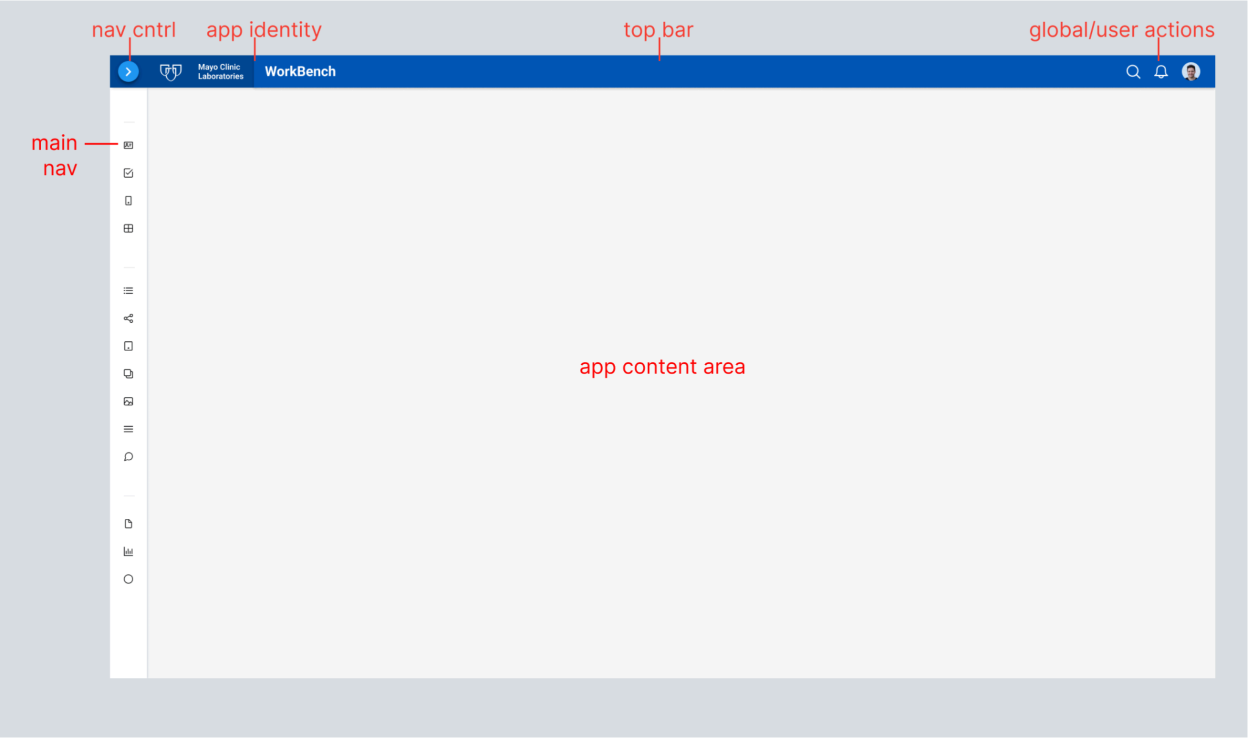
How to choose spacing

When deciding on what spacing to use, consider this:

- Use **less space** between small components, or components that share a close functional relationship.
- Use **more space** between large components, or between components which are less functionally related.

Screen Regions

In a UI layout, users expect to find certain types of content in certain areas. We call these areas screen regions. It is especially important for these zones to be consistent across devices and adapt across breakpoints.



- **Top bar** - Consistent across all GBS applications for use of color, height, identity application and navigation control
- **Side bar** - slim navigation recommended if user requirements allow. This saves valuable space for primary tasks of the application

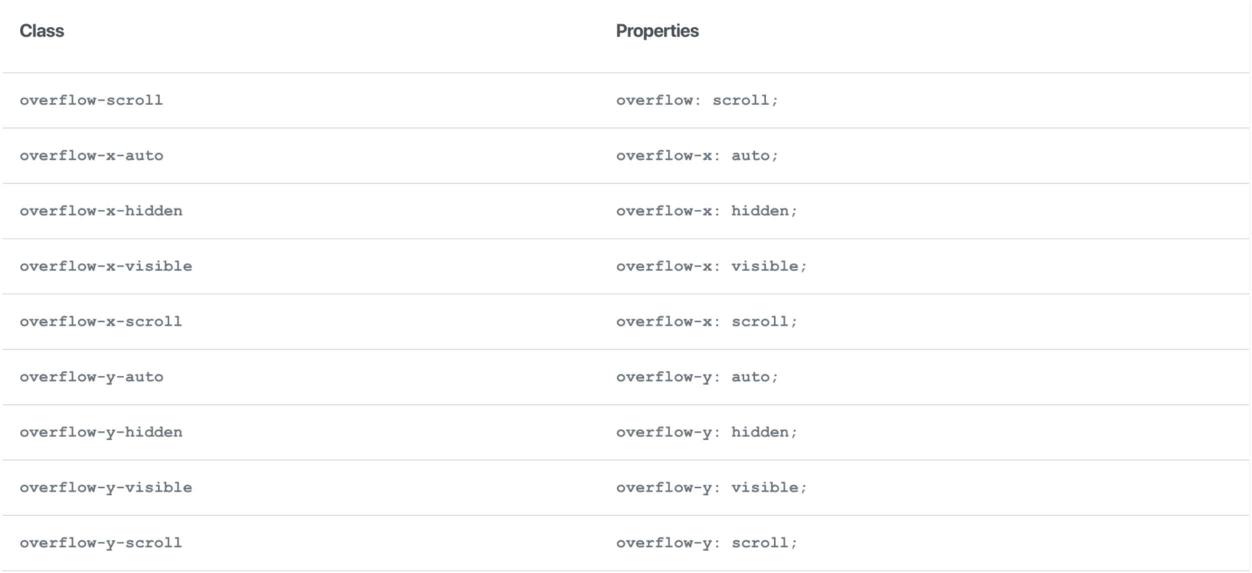
- **Main content area** - this area will be mixed use in terms of components–divided between secondary navigation, data tables, pop-up information etc.
- **Global actions or user-specific actions** - this area should be reserved for global actions that will affect the whole application or for actions by the specific user who is currently logged in.

Scrolling

As part of the PrimeNG framework, scrollbars appear automatically when content falls below the frame or boundaries of the component. In some cases, this can lead to users being unaware of content out of the frame. Consider other ways to to indicate content out of view, such as pagination controls.

Text Overflow

PrimeNG provides multiple classes that control how text behaves when it exceeds the boundaries of its container..



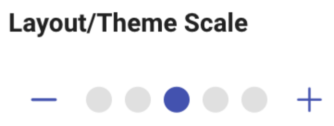
Class	Properties
overflow-scroll	overflow: scroll;
overflow-x-auto	overflow-x: auto;
overflow-x-hidden	overflow-x: hidden;
overflow-x-visible	overflow-x: visible;
overflow-x-scroll	overflow-x: scroll;
overflow-y-auto	overflow-y: auto;
overflow-y-hidden	overflow-y: hidden;
overflow-y-visible	overflow-y: visible;
overflow-y-scroll	overflow-y: scroll;

Most components use overflow-auto.

Text overflow: https://www.primefaces.org/primeflex/textoverflow

Component sizing

The Ultima theme provides scaling options for the interface depending on the requirements of the specific application.



Usage Tips

- Many of the GBS applications contend with the problem of very dense information display. We recommend taking advantage of these scaling options when it makes sense.
- Strive to maintain a balance between clear data display and effective use of space
- Establish a baseline size to dictate the size of components you're going to reuse for the whole app.
  - Baseline size: one of five sizes of "Component Scale". Please see that in the gear icon on the right side of the PrimeNG website. (http://primefaces.org/primeng/setup)
  - E.g. recommended size, table, button, badge.
- Use consistent sizes for components within the same level of hierarchy.
  - Some components will have size options, e.g. button
- Try to use the least amount of visual information possible that still communicates the function of the component to the user.

Reduce Visual Noise

We can change many parameters to make an item visually stand out. This technique is used to guide user actions, but if overused, it can cause distractions. Treat user attention as a valuable and limited resource. Each piece of visual content is competing for a very limited amount of user attention. Thus, we need to be judicious in usage of techniques that make items visually stand out.

Items should stand out when a user needs to:

1. Take a primary action
2. Be notified of an error
3. Address an error
4. Find or navigate to a specific item

Use these techniques to help content stand out:

1. Choose saturated colors or use the designated primary color
2. Display content over a large area of color
3. Display a 100% of a color over a tint of the same color.
4. Display content at the same hierarchical level (flat hierarchy)

Use these techniques to make content stand out less:

1. Use plain text
2. Choose neutral or tinted colors
3. Display content over a small area of color
4. Make content available on demand (for example, add a “More” button)

A common mistake is to make everything stand out. When everything stands out, nothing stands out. To avoid this, assign hierarchies to content. Each piece of content should be assigned a priority (most important to least important), and techniques that call for more or less visual attention should be used accordingly.

Atomic Design

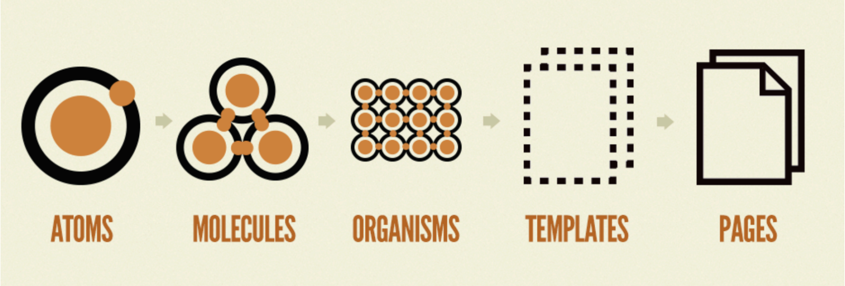
There are many foundational concepts that underlie the decisions made in UI design. If you are looking for a resource that explains the framework and foundations for UI/UX design, Atomic Design Methodology provides an excellent introduction to one of the most important concepts, *the hierarchy of interface design systems*.

Atomic Design Methodology is from *Atomic Design*, an online book by Brad Frost. The author illustrates how to put together the building blocks of UI design, walking the reader from the 'atom' level (the smallest functional unit) to the full page the user sees on the screen.

Below, we have selected text and a picture from *Atomic Design Methodology* to briefly summarize Frost's description of UI design components and how they are used together:

(1) **Atoms:** "basic HTML elements like form labels, inputs, buttons, and others that can't be broken down any further without ceasing to be functional"
(2) **Molecules**: "groups of UI elements that function as a unit"
(3) **Organisms**: "relatively complex UI components composed of groups of molecules and/or atoms." For example, a header banner that consists of the name of the application, a menu, a search bar, and a user profile icon.
(4) **Templates**: "page-level objects that place components into a layout and articulate the design's underlying content structure." For example, the overall layout of a page.

(5) **Pages**: ”specific instances of templates that show what a UI looks like with real representative content in place.” Essentially, it is a template filled in with real content.



UI ——————————————————> **UX**

Image Credit: Atomic Design Methodology

Components

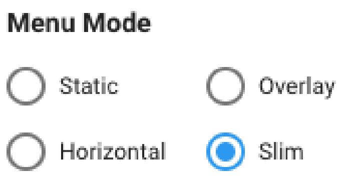
Introduction

All components in this section are selected from the PrimeNG or Ultima library and are presented in their original form (e.g., no modifications or customizations). When using the elements listed below, refrain from making any modifications or customizations.

Main Navigation Menu

The main navigation of the application should reflect the primary verticals of the application.

The Ultima theme includes several options for style and placement of the main nav menu.



Menu Mode

○ Static ○ Overlay

○ Horizontal ● Slim

Recommendation

In general, we recommend the **slim style** as it will save vertical space for data presentation. However, as we cannot foresee all use cases or user responses/requirements, the other styles (such as the horizontal nav bar) may also be used as makes sense. Again, consistency is key. If horizontal menus are used for some apps, ensure that they are styled consistently across those apps.

States

We recommend adherence to all standard state presets in PrimeNG. Components that follow this rule include, but are not limited to:

1. Button
2. Page tab
3. Group tab
4. File Uploader

Button

Introduction

Definitions

A button allows the user to perform an action. They are designed to have visual emphasis, and therefore they call attention to the task.

Dos and Don’ts

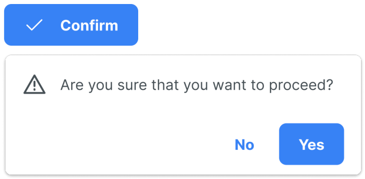
DO:

- □ DO capitalize the first letter of each word in a button (e.g., “Upload Form”).
- □ DO use active verbs (e.g., instead of “Soft Interface,” consider “Choose Soft Interface”)
- □ DO minimize the length of text within a button. Keep the wording succinct.
- □ DO assign hierarchy (most important to least important function; main tasks versus subtasks) when working with more than one button. We have included checklists/instructions to guide your design decisions related to this topic (See Primary, secondary, and tertiary buttons and Sizes).

DON’T:

- □ DON’T allow button text to exceed 20 characters.
- □ DON’T display both icon and text if the icon is self-explanatory (e.g., instead of 🔍 Search, consider 🔍 or Search). You may need to ask the users if the icon is sufficient for communicating the button’s function.

  The rationale for this is that displaying both an icon AND text has a specific meaning in the GBS applications. Icon AND text denotes that a dialog box will open when the button is clicked. Here is an example (see 'Custom Components and Blocks - Dialog'):



Are you sure that you want to proceed?

Options and Usage

Primary, secondary, and tertiary buttons

Primary, secondary, and tertiary buttons are used to distinguish the importance of the actions involved in a task. Without them, users may be confused, perceiving all actions as equally important.

Primary, secondary, and tertiary buttons have variations in emphasis. The primary button has the **most emphasis** and is therefore assigned to the **most important action**. The tertiary button has the **least emphasis** and is therefore assigned to the **least important action**.

We have developed the following checklist to guide you when choosing between primary, secondary, and tertiary buttons:

□ If there is only one button in the functional block, choose the primary button. The remaining checklist items apply to instances where there is more than one button.

□ Assign hierarchy to each button based on functionality. Style the buttons so the hierarchy of their importance is clear.

The **most** important function will receive the **primary** button style, the **second** most important function will receive the **secondary** button style, etc. For example, in a button set that includes “Submit” - “Save” - “Create New”, the primary button should be assigned to “Submit.” The secondary button should be assigned to “Save.” And finally, the tertiary button should be assigned to “Create New.”

☐ When assigning hierarchy, designate one (and only one) button as the primary button. It is fine to have more than one secondary or tertiary button.

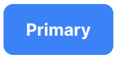
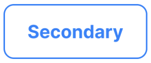
☐ Keep the number of buttons to three or fewer. When you find yourself making a set of **> 3 buttons**, consider another type of component.

A large button set (> 3 buttons) can often be condensed into a single, more advanced functional component, such as a tab. To help decide which component to use, think about functionality (see next point).

☐ Check that each button represents a standalone function that indicates a single action. One way to evaluate this is to ask whether the buttons within a set share a relationship. **If they share a dependent relationship with one another, consider using a different type of component.** Here are some examples:

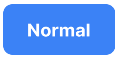
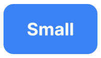
- ○ Mutually exclusive relationships:
    - ■ Instead of a “Yes”-“No” button set, consider a toggle.
    - ■ Instead of a “GRCh37”-”GRCh38” button set, consider a dropdown or radio button.
    - ■ Instead of a “Week”-”Month”-”Quarter”-”Year”-”Year to Date” button set, consider a time picker.
- ○ Non mutually exclusive relationships:
    - ■ Instead of a button set where each button acts as an on/off switch, consider a checkbox.

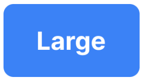
- For instance, where you want to indicate there are multiple choices, consider a tab menu.

	**Primary button:** Button > Button > Basic Primary buttons show the preferred course of action for a user in a given flow. To help maintain consistency throughout the Web App, all actions assigned with the highest hierarchy should be represented by the Primary button. If there is only one button in the functional block, choose the primary button.
	**Secondary button:** Button > Button > Outlined Buttons Use Secondary buttons for actions that have lesser importance. For example, in a button set that includes “Save”-”Cancel”, “Cancel” would be assigned the Secondary button. One could also choose to use a tertiary button here in place of the secondary style.
	Tertiary button: Button > Button > Text Buttons Tertiary buttons are assigned to the least important, but still necessary functions. They are great for minor journeys like viewing terms and conditions. They have the same appearance as regular text.

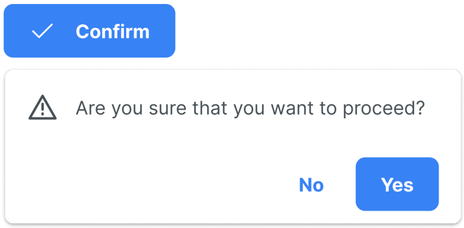
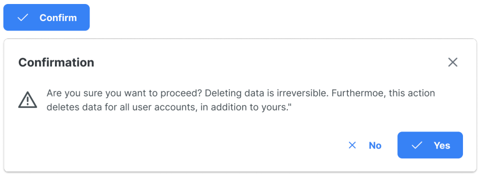
Sizes

In most cases, we recommend that the ‘normal’ sized button is used. In special cases, where tasks are complex and can be subdivided into main tasks and subtasks, we recommend the addition of ‘small’ sized buttons.

	**Button > Button > Sizes > Normal** By default, use the ‘normal’ sized button. These buttons represent the main function in a task (e.g., ‘Submit’).
	**Button > Button > Sizes > Small** The ‘small’ sized button can be used alongside the ‘normal’ sized button to indicate hierarchy within one task. In these cases, hierarchy should be established by distinguishing actions as main tasks versus subtasks. The **‘normal’** and **‘small’** sized buttons should be assigned to actions associated with the **main task** and **subtask**, respectively. The most common use case for the ‘small’ sized button is when the user needs to apply actions to a table. Here are some examples: - When a user needs to apply filters to a table, a small ‘Apply’ button placed above the table would be appropriate. - When a user needs to view more details related to a row of data, a small ‘View Details’ button placed within the row would be appropriate. In the above example, **‘normal’** sized buttons are assigned to actions related to the **main** tasks (e.g., browsing, viewing, and adding data), **‘small’** sized buttons are assigned to actions related to the **subtasks** (e.g., applying filters to the data, viewing more details related to a row of data).

	It is ok to use ‘small’ sized button(s) without the presence of ‘normal’ sized buttons.
	**Button > Button > Sizes > Large** Due to the large amount of data generally displayed on the screen, we recommend that you REFRAIN from using large buttons.

Confirmatory actions

 Are you sure that you want to proceed?	**ConfirmPopup** Use this option when the button represents an action that is: 1.) Difficult to undo/reverse (e.g., deleting a large amount of data). 2.) Absolutely critical and has no room for error (e.g., signing off on a clinical interpretation). In special cases where you might want to communicate to the user what types of things should be considered prior to pressing the button, consider using the ConfirmDialog option instead (see below).
 Confirmation Are you sure you want to proceed? Deleting data is irreversible. Furthermoe, this action deletes data for all user accounts, in addition to yours."	**ConfirmDialog** Use this option when the button represents an extremely important action that requires serious consideration before proceeding.

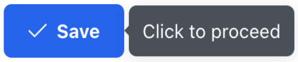
	This option allocates text space for providing more information to the user. It is a great way to communicate to the user what types of things should be considered prior to pressing the button.

Behaviors

Tooltips

Tooltips can be used to communicate a button’s action. Here are questions that will guide the decision on whether or not to use a tooltip:

- ☐ Is the text within a button more than 20 characters?
- ☐ Do you need more space to communicate the button’s action?

	**OVERLAY > Tooltip > Button** When the user hovers over the button, a tooltip with additional text will appear. Use ‘Tooltip’ if the button action requires additional explanation. The tooltip position is dependent on the component. **For buttons**, it should be kept at its default position (right).

Tag

Introduction

Definitions

Use tags to highlight **snippets of key information** to users. For GBS applications, example snippets would be status (e.g., completed, pending, or unreviewed) or data type (e.g., clinical).

When to use?

Consider using tags when:

- □ You need to attract the user's attention. When considering tags, ask the question: *Is this content important enough to warrant extra user attention*? If the answer is 'no,' use plain text.

- □ Content can be conveyed in a concise and pithy manner (~1 word).

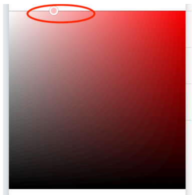
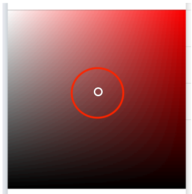
Dos and Don'ts

DO:

- □ DO keep content to ~1 word.

- □ DO include only text and no imagery in the tag.

- □ DO assign content to a specific color. For each unique piece of content, there should be one corresponding color. The background color and the text color should be selected in pairs as follows:

Choice	Background color		Text color	
BLUE	#B3E5FC	🟦	#23547B	🟦
ORANGE	#FFD8B2	🟧	#805B36	🟫
GREEN	#C8E6C9	🟩	#256029	🟩
YELLOW	#FEEDAF	🟨	#8A5340	🟫
RED	#FFCDD2	🟥	#C63737	🟥
PURPLE	#ECCFFF	🟪	#694382	🟪
If you would like to customize tag colors, select the same hue for the background and the text	Choose from this area		Choose from this area	

DON'T:

□ DON'T allow text to take up a long length of space. Use short and pithy, and not lengthy and complex, messages.

Example: LENGTHY OR COMPLEX INFORMATION

□ DON'T allow text to overflow into a new line.

Example: TEXT
OVERFLOW

□ DON'T use a gray tag to convey relatively less important information (e.g., to indicate data is 'unavailable'). In this case, 'unavailable' does not warrant a tag and should be written as plain text.

This is a common mistake, where the logic is that the gray color signifies ‘unimportant.’ However, the use of tags, regardless of its color, brings attention to the content.

Options and Usage

BLUE	**MISC > Tag > Tags** We recommend that you use only the tag type ‘Tags.’ We do NOT recommend usage of tag types ‘Pills’ and ‘Icons.’ The addition of rounded edges and imagery adds visual noise. If you feel called to use rounded edges (‘Pills’), perhaps you are looking to highlight content within a circular “tag” instead. A common example is highlighting numbers. In this case, consider using a ‘Numbers’ badge.

Badge

Introduction

Definitions

Badges are used to indicate numbers. They are usually small and positioned in the upper right hand corner of an icon. Or, they are standalone (not associated with an icon).

When to use?

We recommend that badges are used **only to communicate numbers**. For example, use a badge when you need to communicate to a user that they have 2 unread messages.

Dos and Don't

DO:

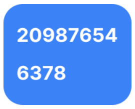
- □ DO use badges to indicate number

- □ DO use only red and blue badges. Most badges should be blue. (See Badge > Options and Usage > Color).

DON'T:

- □ DON'T allow the numeric text to overflow into the next line. If you encounter overflow, consider rounding to remove the decimal or using scientific notation.

Example:



Options and Usage

Color

We recommend only using red and blue badges, since more colors would add visual noise that may overwhelm the user.

- Use red badges when immediate attention is needed (e.g., to indicate the number of messages that users need to address ASAP).

- Use blue badges in all other cases. Most badges will be blue.

Usage

2	**MISC > Badge > Numbers** Use the ‘Numbers’ option in line with text, placed immediately after a segment of text. In these cases, you are using the badge as typography.
🔔 2	**MISC > Badge > Positioned Badge** The most common usage of the ‘Positioned Badge’ is when you are notifying the user the number of alerts or messages they have. When choosing the icon for the positioned badge, the meaning of the icon MUST be self-explanatory. If it is not, use the ‘Numbers’ option (no icon) instead.
Primary 8	**MISC > Badge > Inline Button Badge** We recommend that you REFRAIN from using the ‘Inline Button Badge,’ which is when a badge is placed inside a button. This option creates unnecessary visual noise. Instead of the badge, use plain text + badge (‘Numbers’ option). For example, the image on the left should be: ‘Primary 2’.

Page Tab

Introduction

Definitions

Page tabs are used to organize large amounts of content. Content is grouped and displayed on separate tabs. Page tabs allow for quick navigation between views, where a user can access different sections of content without leaving a page.

In most cases, page tabs will be used to separate content within the same task block. Content under tabs should be related to the same overarching task, and together, they should form one coherent unit.

Use page tabs when a task block is associated with a large amount of content. Instead of having a massive wall of content, divide the content into categories and assign each category to a page tab.

A common mistake is choosing to use group tabs instead of page tabs. The key deciding factor that determines whether you should use page versus group tabs is the **number of hierarchical levels** associated with the tab headers.

Dos and Don'ts

DO:

- □ DO use page tabs when a task is associated with a large amount of content. Divide large amounts of content into categories and assign each category to a tab.

- □ DO ensure that all tabs share some similarities. While the content for each tab is different, try to provide similar data structures, browsing methods and actions within each tab.

- □ DO ensure that all tabs belong to the same task. If they don't, consider using the Group Tab component instead.

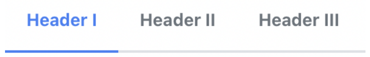
DON'T:

- □ DON'T Insert images in the page tab headers.

Options and Usage

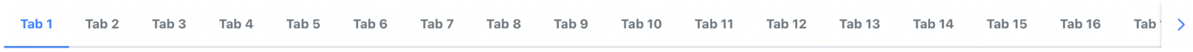
Panel > TabView > Default

We recommend 'Default' for most cases. Consider using other options in special cases, as described below.



Panel > TabView > Scrollable

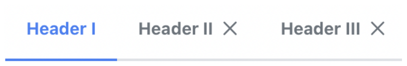
Use the 'Scrollable' option when the total length of all tabs exceeds the optimal width of the content in the tab page.



Panel > TabView > Closable

The 'Closable' option allows the user to remove a tab.

Generally, we recommend that you REFRAIN from using the 'Closable' option. The exception is when the user must make crucial comparisons between tab pages.



Panel > TabView > Programmatic and Custom Headers

Please refrain from using the ‘Programmatic’ and ‘Custom Headers’ options. The addition of redundant text and iconography adds visual noise.

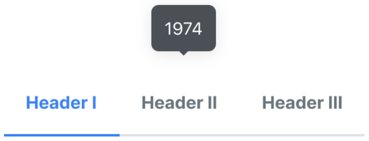
Behaviors

Tooltips

Panel > TabView > Tooltips

Use ‘Tooltips’ when you anticipate that a first-time user may have difficulty in understanding what the tab will show. This often occurs when the header text needs to be abbreviated.

The tooltip position should be kept at its default position (top).



Group Tab

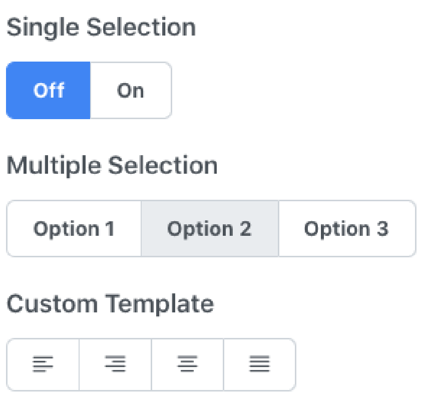
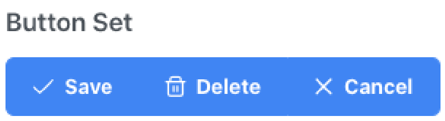
Introduction

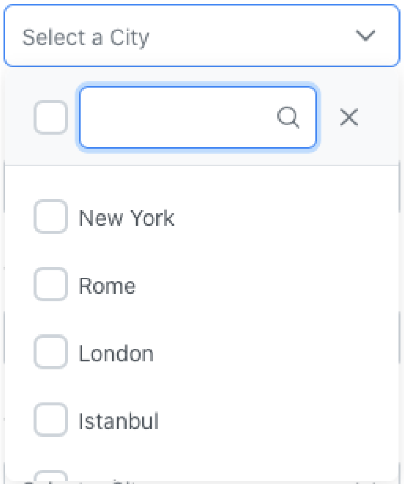
Use these components to navigate, filter or perform an action related to content within a component.

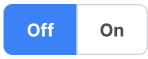
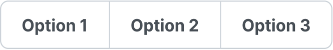
There are three very similar components within the PrimeNG catalog:

- **SelectButton** http://primefaces.org/primeng/selectbutton
- **ButtonSet** (http://primefaces.org/primeng/button)
- **MultiSelect** (http://primefaces.org/primeng/multiselect).

Options and Usage

SelectButton	 Single Selection Off On Multiple Selection Option 1 Option 2 Option 3 Custom Template	Use this component for controlling display of content within a component.
ButtonSet	 	Use this in cases where the user takes action within a component or globally for the whole app. Can be combined with a confirmation dialogue.

MultiSelect		Use when selecting among multiple variables. This can also be used for filtering display of information. Can be used in combination with tags to remind the user of filters set.

	**Form > SelectButton > Single Selection** Use this as a secondary general filter below the page tab hierarchy to control the dataset that displays in a table, for example. A toggle component may also be used for the above application.
	**Form > SelectButton > Multiple Selection** Use the 'Multiple Selection' option when users need to input data into a form (in contrast to inputting data into a table).

Common Mix-ups

Page Tab v. Group Tab

Use the 'Page Tab' component for page-level navigation or data parsing.	Use the 'Group Tab' within smaller components.

Select Button v. Input Switch

There are multiple ways of displaying options when users have two choices.

Use a Select Button if this component stays with other similarly styled components.	Use an 'Input Switch' (also called a toggle) if the choices are binary data type.

Select Button v. Dropdown

Use a 'Select Button' when you have a limited number of choices (<5 choices). In this case, it is okay to display all of the choices, since it will not take too much vertical space.	Use a 'Dropdown' when you have a larger number of choices (5-8) and you don't have the vertical space to display all of the choices. Be aware that if you have more than 8 choices, scrolling can become cumbersome.

Input

Introduction

Definitions

‘Input’ encompasses any field that a user will input content into. Sometimes, this occurs in a few standalone input fields. Other times, many inputs are combined to make a form for the user to fill out.

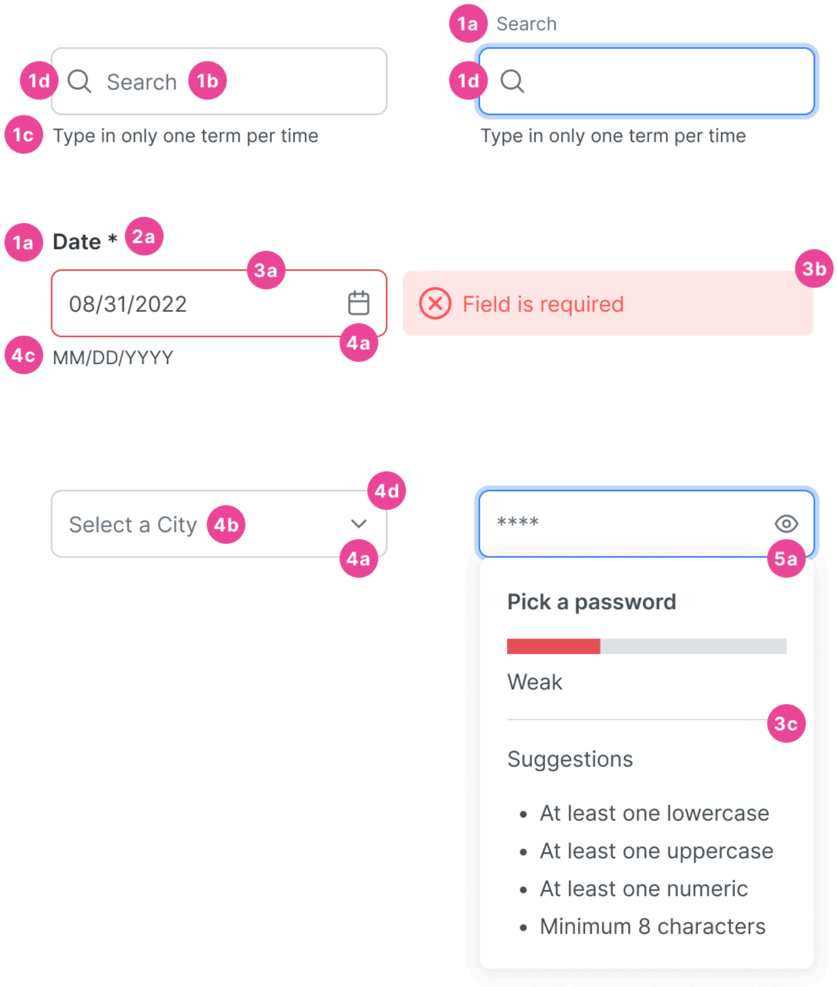
There are many types of input fields:

- Text boxes (both single line or large free text boxes)
- Search bars
- Username and password inputs
- Calendars to input dates
- Dropdowns (single and multi select). For the purposes of this documentation, we describe dropdowns in a separate section (see Dropdown and ListBox)

Using the input options provided in this section and general principles about layout, the input fields will create a unified look and feel as a whole section, and the user will perceive them together as a single task block. In this case, the task block is a form for the user to fill out.

Elements of Input Fields

Any single input field can be broken down into multiple elements. These elements can be added in or out, depending on requirements. Below, we list the elements and their associated purpose, in order of most to least important:



Search

Type in only one term per time

Search

Type in only one term per time

Date *

08/31/2022

MM/DD/YYYY

Field is required

Select a City

****

Pick a password

Weak

Suggestions

- At least one lowercase
- At least one uppercase
- At least one numeric
- Minimum 8 characters

	Purpose	Element
1	Introduce and define the content that the user needs to input	Always included: 1a. Label (e.g., ‘Username’), either static or floated. Optional add-ons: 1b. Placeholder (e.g., ‘m105864’) 1c. Help text (use ‘Tooltips’) 1d. Icon at left (only for search bar)

2	Communicate to the user that input is necessary in order to proceed	2a. Asterisk (e.g., *Electronic Signature)
3	Communicate to the user that their input was invalid and that they need to revise their input	3a. Red stroke 3b. Inline alert 3c. Suggestions
4	Prompt of expected data type	4a. The icon on the right side (e.g., the date icon) 4b. Placeholder 4c. The help text 4d. Dropdown box
5	Special function	5a. Visibility control

Options and Usage

Baseline Options

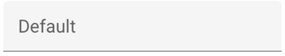
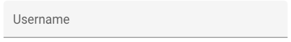
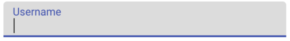
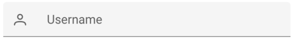
For the most part, the GBS applications will use one of two baseline options:

- In general, choose ‘Basic.’
- When your page includes a long form with many input fields, consider the ‘Float Label’ option, and apply the Layout & Spacing rules. This option keeps the format clean.

In the special cases listed below, refer to “Specialty Options”:

- Search bar
- Extra long text (i.e., free text box)
- Numbers with specific formats

- ○ Commas and decimals
- ○ Formatted numbers (e.g., 99-999999, dates such as 05/13/2022

 Default	**Form - InputText > Basic** This is the basic text field for entering data. When using this option, the label can be placed in one of two places: 1) At the left of the input field. 2) On top of the input field. The text inputted by the user should always be left aligned.
First appears as:  Username When user types in the input field, location of label changes to:  Username	**Form > InputText > Float Label** shows options for free text input only. **Form > FloatLabel** shows more options (inputs for free text, calendar, dates, drop downs, etc.). The 'Float Label' option eliminates labels. Use this option when you want to keep the format clean. Refrain from using 'Float Label' when the label is long (generally >2 words). When working with longer labels, keep grid layout and alignment in mind.
 Username	**Form > InputGroup** InputGroup is a class within PrimeNG that allows you to group text, buttons, and images with input fields.

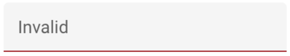
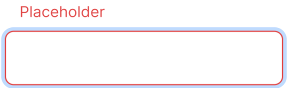
	If you choose to use this style, verify your reason for doing so and then, use it consistently across your application. For example, you may want to use this style for Admin Settings pages where you have many form elements where additional visual information may help with comprehension and scanning. Do not mix and match this style with regular input form styles.

Invalid Text

These options indicate to the user that invalid text has been inputted. There are two options, and the choice is dependent on the Baseline Type you chose.

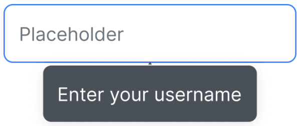
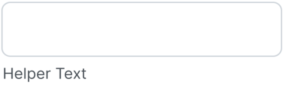
- If Baseline Option = **‘Basic**,’ choose **‘Invalid**.’
- If Baseline Option = ‘**Float Label**,’ choose ‘**Invalid with Float Label**.’

If you need to communicate to the user why the text was considered invalid, add an Inline Message (see ‘Messages - Inline Message’ section).

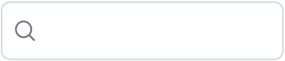
	**Form > InputText > Invalid** The text field will appear as invalid as soon as the user clicks on any space outside of the text input box.
	**Form > InputText > Invalid with Float Label** The text field will appear as invalid as soon as the user clicks on any space outside of the text input box.

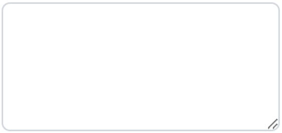
Tooltips

Sometimes, users may need a prompt or additional guidance in order to understand what text they need to input. Although PrimeNG addresses this problem with a ‘Help Text’ option, we ask that you **REFRAIN from using ‘Help Text.’** Instead, we recommend that you use ‘Tooltips’ instead.

	**Overlay > Tooltip > Positions - Bottom** We recommend that ‘Positions’ is set to “**bottom**,” since this avoids covering the label and minimizes eye movement (i.e., the distance between the instructional text and the input field).
	**Form > InputText > Help Text** We recommend that you REFRAIN from using the ‘Help Text’ option. Instead, we recommend that you use ‘Tooltips’ instead (see item above). The rationale for this decision is that if multiple input fields use the ‘Help Text’ option, all instructional text at once. This creates visual noise that could be avoided, since the user only needs to see one instructional text at any given moment.

Specialty Options

	**InputText > Left Icon**

	Use this option when users need to input text to search a database (i.e., search bar). Use the ‘Left Icon’ option and use a magnifier icon.
	**Form > InputTextArea > Default** Use this option when users need to enter extra long text. When using this option: ● Pay extra attention to layout and spacing. In most cases, align the width/length of a text area with components nearby. ● Align the text area with other nearby components to the left.
************	**Form > Password > Show Password** Use this option when the user needs to input a password.
Integer Only 42,723	**Form > InputNumber > Numerals** Use this option when the user needs to input numbers with commas or decimal points.
99-999999	**Form > InputMask** Use this option when the user needs to input numbers with a specific format (e.g., 99-999999, dates such as 05/13/2022)

Messages

Introduction

Definitions

Messages prompt users to acknowledge status and take further action. They are meant to be noticed.

Keep messages concise. Additionally, for the Info, Warning, and Error messages, write clear messages for the IT team. The IT team should be able to understand the message without having to contact the user who encountered the error.

Options and Usage

Static notifications

Messages > Messages > Static Content

The 'Static notifications' option stays open at all times.

Usually placed at the top of the page, they are used to display content that a user might need to refer to multiple times.

Use this option sparingly. Static notifications are often overused in places where plain text should be used instead. Here are some characteristics that will help determine if content is important enough to be highlighted as a static notification:

- □ Does the user need to refer to this content multiple times in order to complete a task?
- □ Does the user need to know this information in order to complete a task?

□ Is this content more urgent than a casual FYI?

Always bet on Prime.

Closable alerts

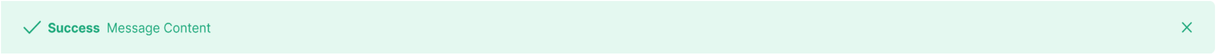
Messages > [Messages](#) > Severities

The ‘Closable alert’ option will disappear when the user clicks the ‘X’ to close it. It tells the user, “We need your attention for a task, and you need to let us know when you’ve completed it.” A common use case is when a user needs to address an error.

PrimeNG grades these messages by severity. From least to most severe, they are: Success, Info, Warning, and Error.

DO use this option for ‘Info,’ ‘Warning,’ and ‘Error’ messages.

DON’T use this option for ‘Success’ messages (e.g., to communicate that a task has been completed, a form has been submitted, etc.). Instead, use the “Toast alerts” option, which automatically disappears after a set amount of time.

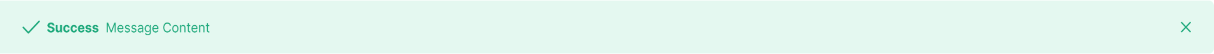

✓ **Success** Message Content ×

Toast Alerts

Messages > [Toast](#) > Severities

The ‘Toast alert’ automatically disappears after a set amount of time. Always use the ‘Toast - Severities’ option from PrimeNG (toast slides out from the top right of the page).

This option is most commonly used for ‘Success’ messages (e.g., to communicate that a task has been completed, a form has been submitted, etc.). Its purpose is to acknowledge and confirm that something important has taken place. It validates for the user that the information has been received or processed.



Success Message Content

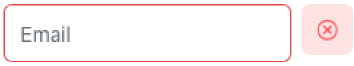
Inline Messages

Messages > [Messages] > Inline Message

The ‘Inline Message’ option is most commonly used in forms, usually to indicate to the user that the text entered was considered invalid.

Use this option alongside the ‘Invalid’ or ‘Invalid with Float Label’ options (see ‘Input - Invalid Text’ section).

The message should permanently display until the user has fixed the input and the backend has deemed it as a pass.



Email

Dropdown and ListBox

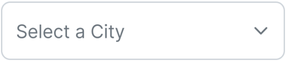
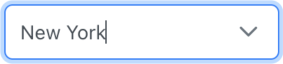
Introduction

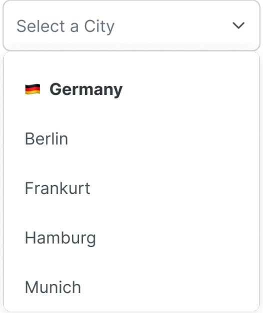
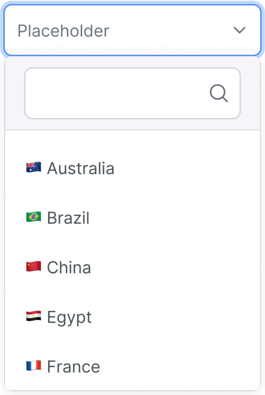
Definitions

Both ListBox and DropDown components allow users to choose from more than one item. PrimeNG provides many variations on these, allowing users to pick one from many or many from many, with additional functionality that can be added on such as type ahead.

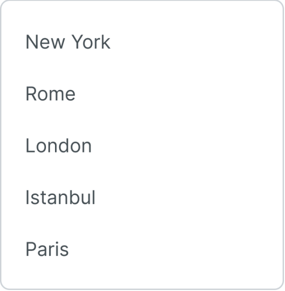
Options and Usage

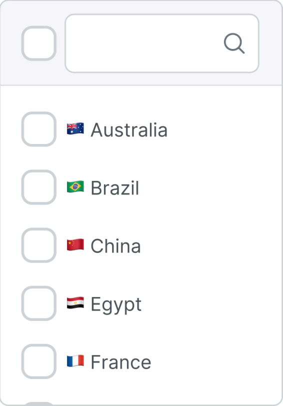
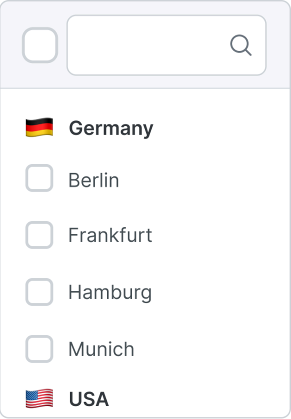
Dropdown

	**Form > Dropdown > Basic** Use the 'Basic' option when you need to standardize or limit the user's input. Unlike entering free text, this option avoids inputting variations in answers for a single item. This is especially helpful for choosing a name for an item.
New York	**Form > Dropdown > Editable** The 'Editable' option is just like the 'Basic' option, except with the addition of allowing users to type in free text. Use this option if a user might need to make slight customizations to the standardized answers. Displaying the existing choices saves time for the users.

 Select a City **Germany** Berlin Frankurt Hamburg Munich	**Form > Dropdown > Group** Use the ‘Group’ option when the list of choices is long. Categorize the items, and display them with the ‘Group’ option.
 Placeholder Australia Brazil China Egypt France	**Form > Dropdown > Advanced with Templating, Filtering and Multiple Selection** Use this option when there are many choices to choose from. This allows the user to search instead of scroll through all of the choices.

Listbox

 New York Rome London Istanbul Paris	**Form > Listbox > Single** Use it for “Basic” and “Editable” dropdowns.

	**Form > Listbox > Advanced with Templating, Filtering and Multiple Selection** Use it for the “Advanced with Templating, Filtering and Multiple Selection” dropdown.
	**Form > Listbox > Advanced with Group, Filtering and Multiple Selection** Use it for the “Group” dropdown by deleting the search box if you like the pattern in the Form - Dropdown - Group. (This might be a mistaken documentation in the PrimeNG guideline)

Scroller

Introduction

Definitions

The scroller allows users to view a small subset of information at a given time. It can be used to display text or data table content when you encounter limited vertical height, and when the content that needs to be displayed exceeds the height of the reading window.

When to Use

The main question regarding usage is - *Should I use 'Scrolling' or 'Pagination'?*

In general, use pagination instead of scrolling - This is because pagination works best when users are looking for specific pieces of content AND they have a general sense of how much/where the information is positioned within the data. Since the data displayed in GBS applications are generally ordered (by date, alphabetical, etc.), users usually have a general sense of where to find their data.

In contrast, scrolling is better suited for casual browsing or smaller data sets.

There ARE some cases where a user may not know where their data will be. Sometimes, a user might not have a frame of reference for where the data they are looking for is. In this case, scrolling would be appropriate.

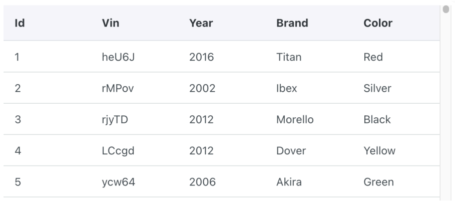
There are also users who prefer scrolling. Ask your users how they like to move through data results. If users prefer scrolling, consider a combination of robust filtering tools and an infinitely scrolling data table.

(Pagination is further described in 'Custom Components and Blocks - Small Paginator.')

Options and Usage

There are multiple options for scrollers. The scroll option you choose is dependent on the type of content you are displaying (e.g., table, plain text, a list from a dropdown menu).

Table Display

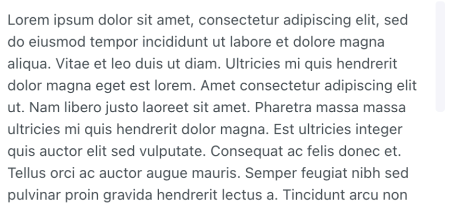


Id	Vin	Year	Brand	Color
1	heU6J	2016	Titan	Red
2	rMPov	2002	Ibex	Silver
3	rjyTD	2012	Morello	Black
4	LCcgd	2012	Dover	Yellow
5	ycw64	2006	Akira	Green

Table > VirtualScroll > both options listed are ok

There are two options listed under ‘VirtualScroll.’ Choose the option that better fits the frontend decisions.

Plain Text



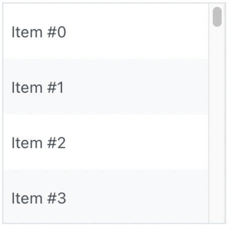
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Vitae et leo duis ut diam. Ultricies mi quis hendrerit dolor magna eget est lorem. Amet consectetur adipiscing elit ut. Nam libero justo laoreet sit amet. Pharetra massa massa ultricies mi quis hendrerit dolor magna. Est ultricies integer quis auctor elit sed vulputate. Consequat ac felis donec et. Tellus orci ac auctor augue mauris. Semper feugiat nibh sed pulvinar proin gravida hendrerit lectus a. Tincidunt arcu non

Panel > ScrollPanel > leftmost option

Use it for displaying plain text that does not fit within the height of the reading window.

There are three options listed on the ‘ScrollPanel’ page. Choose the option on the left (gray scroll bar).

List



Item #0

Item #1

Item #2

Item #3

Data > Scroller > Basic > Vertical

Use this option with a dropdown box.

Loading

Introduction

The loading animation reduces uncertainty associated with waiting for an action to be confirmed. This is important because it refocuses user energy so that they can focus on more crucial tasks. This is especially important for the GBS applications, since there are many tasks a user must accomplish.

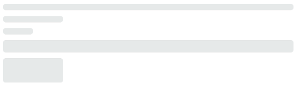
The more you communicate to the users during the wait time, the less anxiety users feel. Here are a couple key points we can communicate through the 'Loading' component:

1.) **Which segments are loading.** Instead of displaying a loading animation over an entire page, display it only over the segment that is loading (e.g., a data table).

2.) **The type of content that will be displayed after loading is complete.** The animation can imply certain types of information. This is accomplished by the 'Skeleton' option. The skeleton foreshadows the type of content by displaying the expected shape of the loaded information.

    This allows users to validate if the content is what they need, as it is loading. If users deem the projected content is not what they need, wait time is reduced.

Options and Usage

	**MISC > Skeleton** We recommend the 'Skeleton' as the first choice option, since it provides more information to the users (see explanation above, 'Introduction').

	Choose the skeleton shape that best matches the loading information. Also, place it in the same position where the loading information will eventually appear.
	MISC > ProgressSpinner > Basic While ‘Skeleton’ is the first choice option, if there are limitations on development time, the ‘ProgressSpinner - Basic’ is an acceptable second choice.

Divider

Introduction

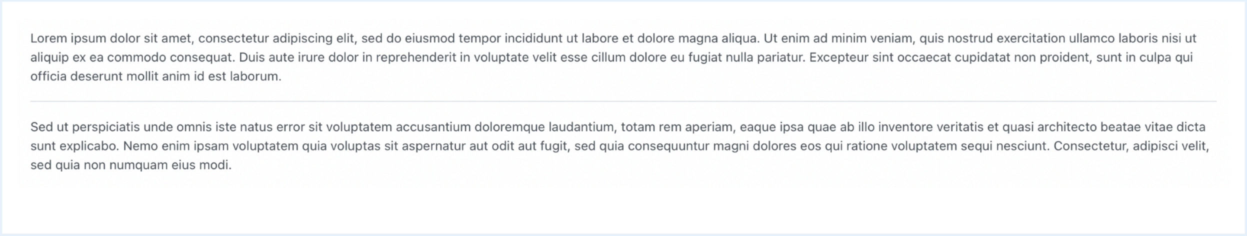
Dividers bring clarity to a layout by grouping and dividing content in close proximity. They can also be used to establish rhythm and hierarchy.

Dividers are especially useful when the vertical space is limited (e.g. in a pop-up dialog window).

Options and Usage

Panel > Divider > Basic

For all dividers, choose the ‘Basic’ option.



Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Sed ut perspiciatis unde omnis iste natus error sit voluptatem accusantium doloremque laudantium, totam rem aperiam, eaque ipsa quae ab illo inventore veritatis et quasi architecto beatae vitae dicta sunt explicabo. Nemo enim ipsam voluptatem quia voluptas sit aspernatur aut odit aut fugit, sed quia consequuntur magni dolores eos qui ratione voluptatem sequi nesciunt. Consectetur, adipisci velit, sed quia non numquam eius modi.

File Uploader

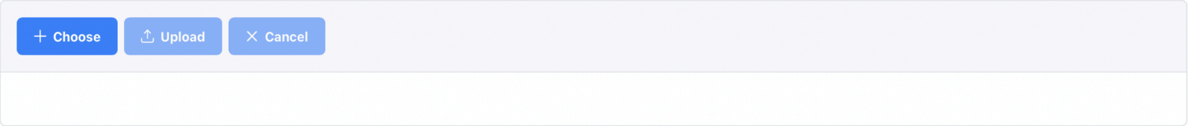
Introduction

The File Uploader component allows the user to upload data (usually an Excel sheet) from their local drive. This component also shows the progress of the upload and provides feedback on the upload status.

Options and Usage

File > FileUpload > Advanced

For all instances where the user needs to upload the file, use the ‘Advanced’ option.



Custom Components and Blocks

Introduction

Unlike in the previous section (‘Components’), the items listed in this section are NOT the original material provided by PrimeNG. Small modifications to PrimeNG library materials have been made in order to fit within the unique contexts of the GBS applications.

This section includes two types of custom items:

1.) **Blocks** (‘Table,’ ‘Insert Content, and ‘Dialog’)
2.) **Components** (‘Breadcrumb’ and ‘Small Paginator’)

The items in this section are listed in order of complexity (most complex at the top, least complex at the bottom).

Approach for Custom Blocks

We highly recommend that you use the Custom Blocks provided in this section. These should be used as templates, meaning we expect that modifications will be made.

Follow the template provided in the ‘Anatomy’ sections, filling components in as you see fit. The template is designed to accommodate the maximum number of components we predict a block would use. Therefore, **remove any component in**

the template that is not applicable to your user interface. Do NOT fill components in the template for the sake of filling empty space.

> *Example: In the 'Table' custom block, we present four different ways to filter data. Most likely, the user will not need all of these components. Remove them as needed.*

What if the templates do not fit my user needs?

If the templates provided do not fit your user needs, it is ok to create your block. This is *NOT the recommended first choice*, but we anticipate that there may be unique user stories that cannot be satisfied with standard templates.

If you do decide to create your own block, here are some pointers that may help make your design decisions:

1.) Refer to the principles in the 'Foundations' section and use the components listed in the 'Components' section. We suggest that you start with the example images, then refer to the written rules as general guidelines.

2.) Creativity and flexibility is expected, due to the chaotic nature of design theory. DO ensure that all design decisions satisfy the general rules outlined in the 'Foundations' section.

3.) As a reminder, pay special attention to the spacing and keep alignment consistent across the entire application.

4.) **Always consider contacting us!** We are happy to offer suggestions at any time. You can find our contact information in the 'Overview - Contact' section.

Approach For Custom Components

We have modified the ‘Breadcrumb’ and ‘Paginator’ components from the PrimeNG library. Start with the PrimeNG components as a basis, and make the modifications described in this document. For each custom component, we have provided step-by-step instructions to make the custom component.

Inspecting Figma code

Figma has an ‘Inspect’ function, which allows you to highlight a selection in the mock-up and view the code. The ‘Inspect’ function will provide code for:

- Spacing and layout
- Color for each item
- Typography settings

Here are the steps for inspecting Figma code:

1.) Open the Figma link (provided in the ‘Figma Inspect Link’ section of each custom block).
2.) Click ‘inspect’ on the right panel.
3.) Find the ‘Code’ section in the right panel. (See red box in picture below)

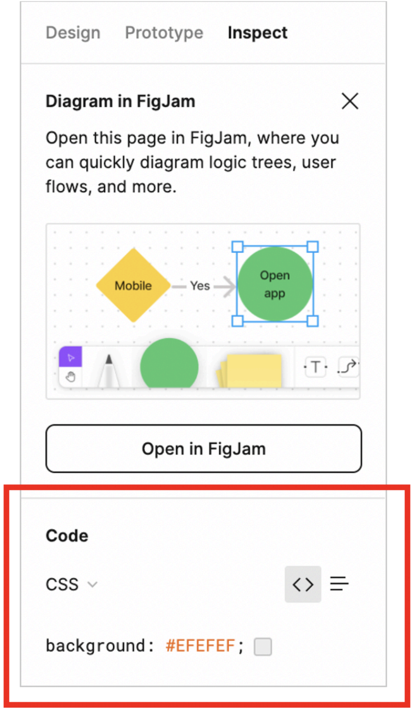


Diagram in FigJam

Open this page in FigJam, where you can quickly diagram logic trees, user flows, and more.

Code

```
background: #EFEFEF;
```

4.) Double-click on an item in the center of the canvas. You will see code appear on the right panel.

5.) **Double-clicking is the trick to selecting items in Figma.** Make a second double-click to view a smaller section of the template.

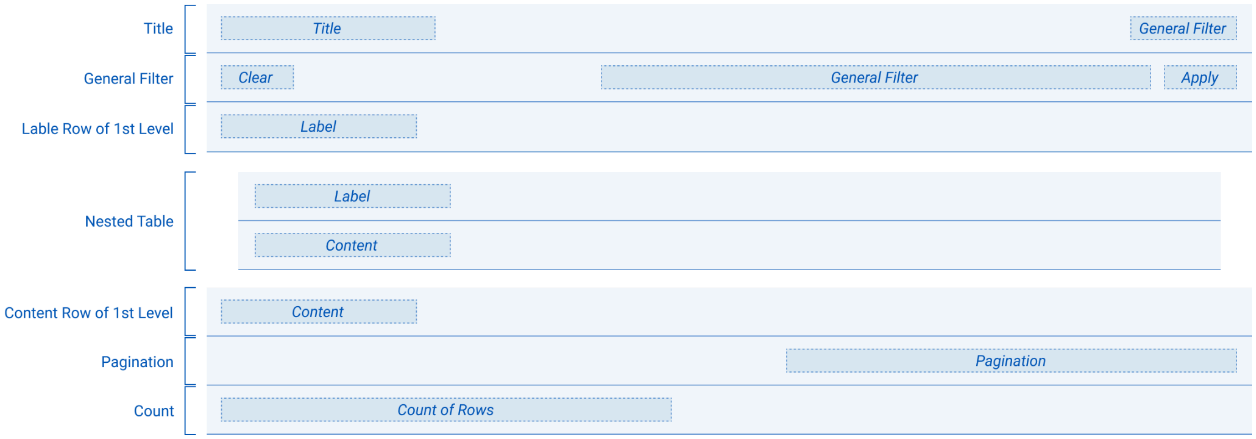
Example: Let's say you want to view the code for the red 'Danger' tag in the 'Table' custom block. Place your mouse over the 'Danger' tag. Make 7 double-clicks. Note that for each double-click, a smaller section of the template will be selected. After 7 double-clicks, the code for the 'Danger' tag will appear in the 'Inspect' panel.

Table

Anatomy

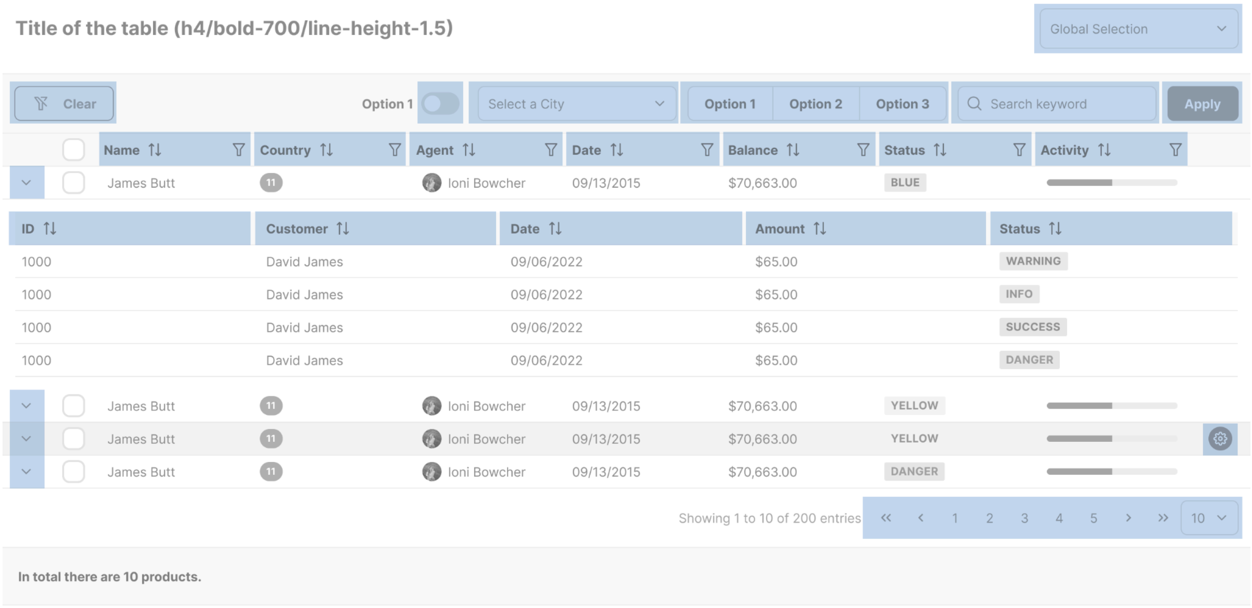
Template

Below is the template for the ‘Table’ custom block. Please note the terms, “Title” and “General Filter,” which will be referred to in the ‘Instructions’ section.



Clickable parts

Due to the complexity of the ‘Table’ custom block, we are including an image that highlights components that are clickable functions (e.g., buttons, drop down menus, sorting, filtering, etc.).



Title of the table (h4/bold-700/line-height-1.5)

	☐	Name	Country	Agent	Date	Balance	Status	Activity
⌄	☐	James Butt	11	Ioni Bowcher	09/13/2015	$70,663.00	BLUE	

ID	Customer	Date	Amount	Status
1000	David James	09/06/2022	$65.00	WARNING
1000	David James	09/06/2022	$65.00	INFO
1000	David James	09/06/2022	$65.00	SUCCESS
1000	David James	09/06/2022	$65.00	DANGER

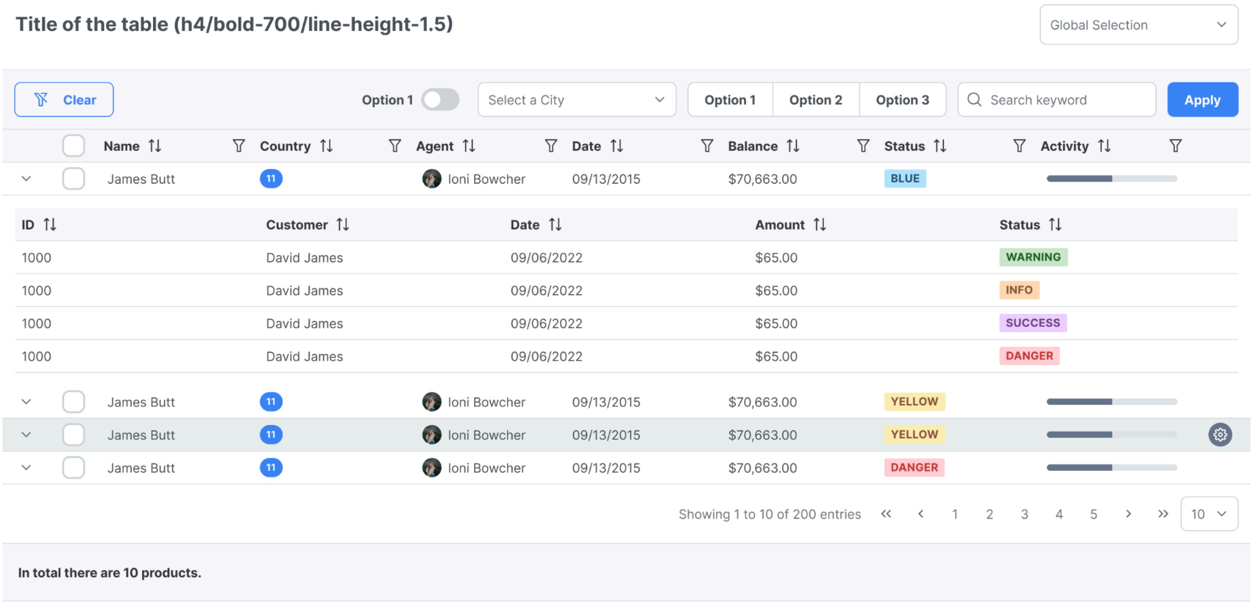
	☐	Name	Country	Agent	Date	Balance	Status	Activity
⌄	☐	James Butt	11	Ioni Bowcher	09/13/2015	$70,663.00	YELLOW	
⌄	☐	James Butt	11	Ioni Bowcher	09/13/2015	$70,663.00	YELLOW	
⌄	☐	James Butt	11	Ioni Bowcher	09/13/2015	$70,663.00	DANGER	

Showing 1 to 10 of 200 entries

In total there are 10 products.

Making the block

Here is an example of the customized 'Table' block:



Title of the table (h4/bold-700/line-height-1.5)

	☐	Name	Country	Agent	Date	Balance	Status	Activity
⌄	☐	James Butt	11	Ioni Bowcher	09/13/2015	$70,663.00	BLUE	

ID	Customer	Date	Amount	Status
1000	David James	09/06/2022	$65.00	WARNING
1000	David James	09/06/2022	$65.00	INFO
1000	David James	09/06/2022	$65.00	SUCCESS
1000	David James	09/06/2022	$65.00	DANGER

	☐	Name	Country	Agent	Date	Balance	Status	Activity
⌄	☐	James Butt	11	Ioni Bowcher	09/13/2015	$70,663.00	YELLOW	
⌄	☐	James Butt	11	Ioni Bowcher	09/13/2015	$70,663.00	YELLOW	
⌄	☐	James Butt	11	Ioni Bowcher	09/13/2015	$70,663.00	DANGER	

Showing 1 to 10 of 200 entries

In total there are 10 products.

Figma Inspect Link

See the “Table” page that you can find on the left panel.

Instructions

Start with the ‘Table - PrimeNG’ material provided by PrimeNG.

The original PrimeNG table provides a good base, so you will NOT need to change any of the following elements:

1. The row that includes column titles, filter icon, sorting icon
2. The checkbox column on the far left column
3. The hovered and focused states (i.e., when the user hovers over a row, it turns gray)
4. The button with the ‘Setting’ icon on the far right column

Changes:

1. Add the ‘Title’ section (see ‘Table - Anatomy’) to the table:
    a. Place this section above the table
    b. Add text on the left (**h4/bold-700/line-height-1.5**)
    c. Add a general filter dropdown selection on the right. The dropdown can be removed if it is not applicable to the use case.
2. Remove the original general filter line and replace it with the customized ‘General Filter’ section.
    a. Add an ‘Apply’ button on the far right.
    b. Add a clear button on the far left.
    c. Add your choice of filters
        i. Select one or more of the provided filters: toggle (also called ‘input switch’), dropdown menu, select button, and search box.
    d. Place all filters and the apply button into one flex group.

e. Place this flex group on the far right.

3. Change the height of the label line to 36px.
4. For small tables, replace the original pagination with the customized, smaller one provided in this document (‘Custom Components and Blocks - Small Paginator’).
5. Set the button with the ‘Setting’ icon to show only when the mouse is hovering over a row.
6. Add a 36px footer (count the number of rows displayed in the current view).

Table > Dynamic

The dynamic table adjusts column width as the user changes the width of their screen. Use this as an underlying setting that is on for all tables.

Table > Size

We recommend that you use the **‘Small Table’ size** as the default size for most tables. In this option, the line heights of rows are 36px.

A ‘Normal Table’ size is appropriate for datasets that:

1.) Do not contain condensed information AND

2.) Are NOT associated with browsing genomic data.

The data must meet both conditions listed above in order to be placed in a ‘Normal Table’ size.

We do NOT recommend using the Large Table size.

Please override the size setting in the original PrimeNG ‘Table - Documentation.’

Table > Row Expansion

Use this option for nested tables.

When using the ‘Row Expansion’ option, make the following changes:

1. Add the arrow icon button
2. Add a blank space in the label row that corresponds to the arrow column
3. Change the width of arrow column to 66px
4. Add a nested table, if applicable

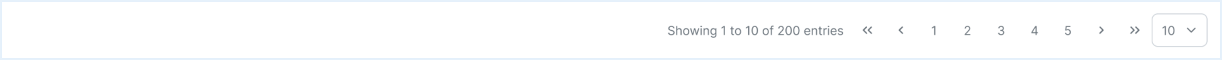
Additional changes

1. Place the arrow icon button column on the far left
2. Change all paddings around the nested table to 14px

Custom Blocks > Small Paginator

If you are using a ‘Normal Size’ Table, use the paginator provided by primeNG (‘Table - Documentation’). Use a paginator only when the height of the table rows exceed the height of the screen.

For ‘Small Size’ tables, use the customized paginator provided in this document, which has been sized to fit the ‘Small Size’ tables (‘Custom Components and Blocks - Paginator’).



Text Overflow > PrimeFlex

Please apply text overflow for the title, the label text, and all text in the cells.

Acceptable Modifications

Below is a list of acceptable modifications for customized tables:

1. The text of the title. We recommend that the title is no longer than one line.

2. Add or remove columns as needed. As you modify the columns, please keep the sequence as follows: arrow icon column → checkbox column → other columns.
3. Add and remove filters as needed. Keep the number of filters minimal - just enough to meet the user's most frequent and typical needs.
4. Change the content in cells as you wish. A few notes:
    a. There are many types of elements and components that can be placed inside cells. They include: plain text, tag, progress bar, buttons (icons, text buttons), forms (input switch), and other selections from the PrimeNG library.
    b. Remember to select components that **stand out less** (See 'Foundations - Reduce Visual Noise')
5. For clickable icons, you can decide between the 'Rounded Icon Buttons' option or the 'Rounded Text Icon Button.'
    a. If you need to drive user attention to the button, choose the 'Rounded Icon Buttons' option.
    b. Otherwise, use the 'Rounded Text Icon Buttons.'

Options and Usage

Scrolling

Please refer to the section, 'Components - Scroller - Introduction." Generally, pagination should be used, but there are some circumstances where scrolling is appropriate.

Grouping

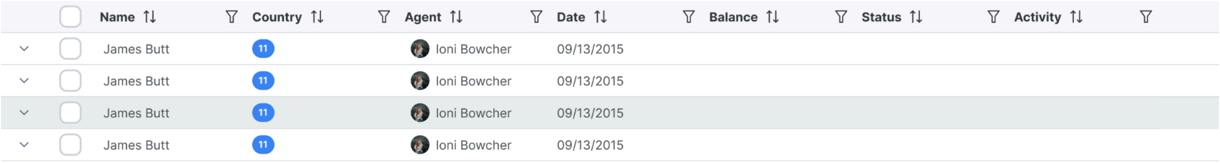
Table > Column Group

Use the ‘Column Group’ option when you need to categorize columns. DO use this option if you can, since this greatly aids the viewer in reading.

Empty Columns

When making a table, always anticipate when the user might encounter an empty column.

These empty columns should be placed on the right side of the table, as pictured below. There should be **no empty columns in the middle of the table**.



	□	Name ⇅	Country ⇅	Agent ⇅	Date ⇅	Balance ⇅	Status ⇅	Activity ⇅
⌄	□	James Butt	11	Ioni Bowcher	09/13/2015			
⌄	□	James Butt	11	Ioni Bowcher	09/13/2015			
⌄	□	James Butt	11	Ioni Bowcher	09/13/2015			
⌄	□	James Butt	11	Ioni Bowcher	09/13/2015			

Editing

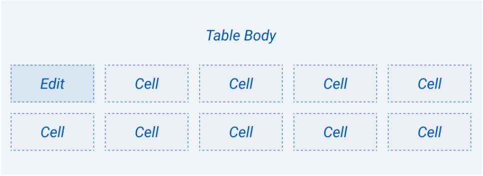


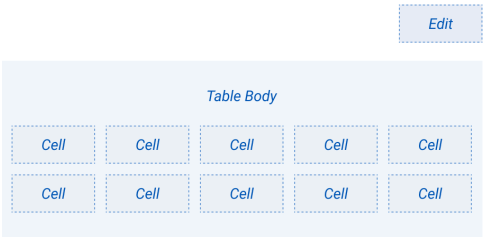
Table Body

Edit	*Cell*	*Cell*	*Cell*	*Cell*
Cell	*Cell*	*Cell*	*Cell*	*Cell*

MISC > Inplace > Basic

This option provides a user to make an *in situ* edit from the ‘view’ mode (e.g., they do not need to open up a ‘edit’ mode).

When the user hovers over the text field, the background color changes to gray. This communicates to the user that the text field is clickable and editable.



Overall Edit

In this option, there are two separate modes, ‘edit’ and ‘view.’ The user needs to click the ‘Edit’ button in order to get to the ‘edit’ mode.

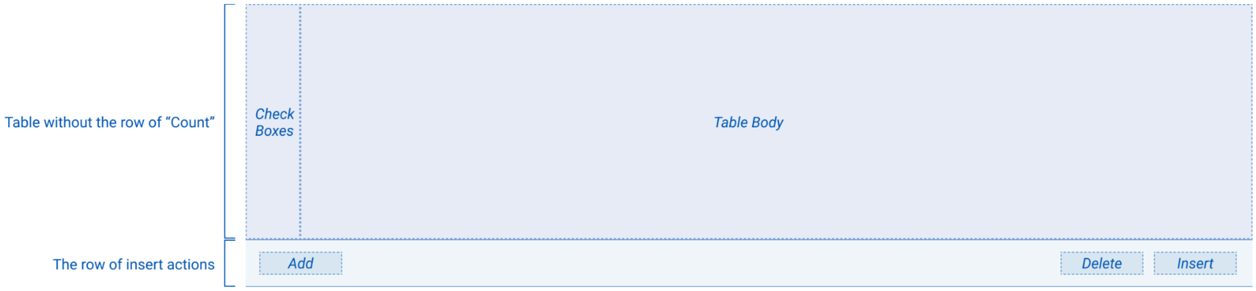
Insert Content

Adding rows of data into a table is a common function in the GBS applications. This section provides instructions for making a block that serves this function.

Anatomy

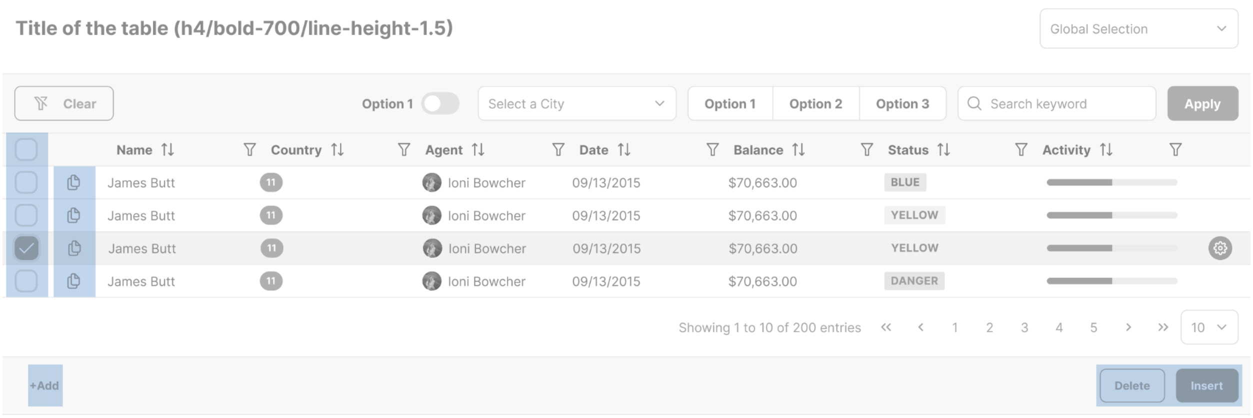
Template

Below is the template for the ‘Insert Content’ custom block:



Clickable parts

Due to the complexity of the ‘Insert Content’ custom block, we are including an image that highlights components that are clickable functions (e.g., buttons, checkboxes)



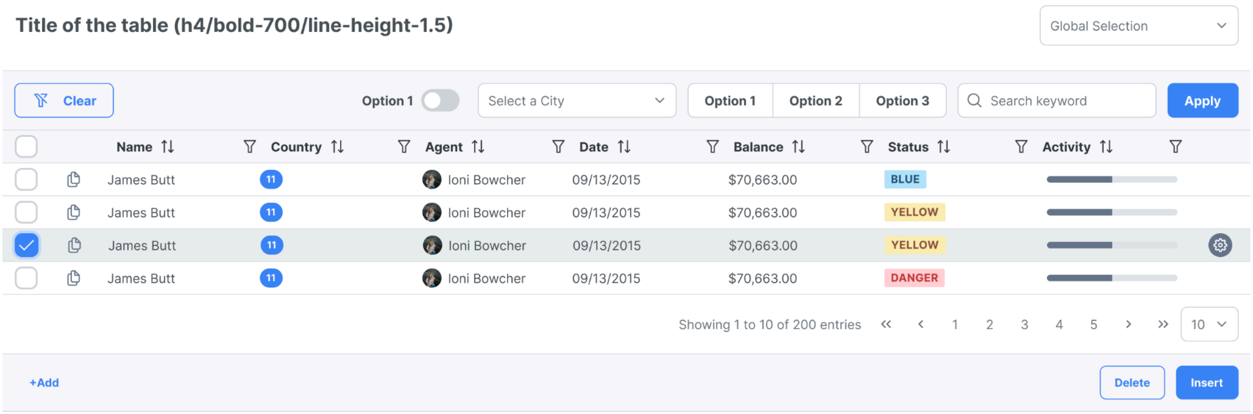
Title of the table (h4/bold-700/line-height-1.5)

□		Name	Country	Agent	Date	Balance	Status	Activity	
□		James Butt	11	Ioni Bowcher	09/13/2015	$70,663.00	BLUE		
□		James Butt	11	Ioni Bowcher	09/13/2015	$70,663.00	YELLOW		
☑		James Butt	11	Ioni Bowcher	09/13/2015	$70,663.00	YELLOW		
□		James Butt	11	Ioni Bowcher	09/13/2015	$70,663.00	DANGER		

Showing 1 to 10 of 200 entries « ‹ 1 2 3 4 5 › » 10

Making the block

Here is an example of the customized ‘Insert Content’ block:



Title of the table (h4/bold-700/line-height-1.5)

□		Name	Country	Agent	Date	Balance	Status	Activity	
□		James Butt	11	Ioni Bowcher	09/13/2015	$70,663.00	BLUE		
□		James Butt	11	Ioni Bowcher	09/13/2015	$70,663.00	YELLOW		
☑		James Butt	11	Ioni Bowcher	09/13/2015	$70,663.00	YELLOW		
□		James Butt	11	Ioni Bowcher	09/13/2015	$70,663.00	DANGER		

Showing 1 to 10 of 200 entries « ‹ 1 2 3 4 5 › » 10

Figma Inspect Link

See the “Insert Content” page that you can find on the left panel.

Instructions

If you want to give users the ability to insert…

- Multiple rows of data at once → follow **steps 1-3**.

- One row of data at a time → skip to **step 4**.
- Multiple rows at once AND one row at a time → follow **all steps**

1. Remove the text inside the row entitled ‘Count’ in the Table Custom Block.
2. Customize the buttons:
    a. Group the secondary and primary buttons.
        i. The primary button should be on the right side of the secondary button.
        ii. Separate the primary and secondary buttons with a 14px gap.
    b. Add the tertiary button.
        i. Place it alongside the other buttons in a flexbox.
        ii. Use the “space between” setting.
    c. Replace the text in the three buttons (see example image above):
        i. Primary button: Insert
        ii. Secondary button: Delete
        iii. Tertiary button: +Add
3. Enable the checkbox actions.
4. Add the ‘copy’ column.
    a. Add a new column. Keep this column title empty.
    b. Change the original icon to a ‘copy’ icon.

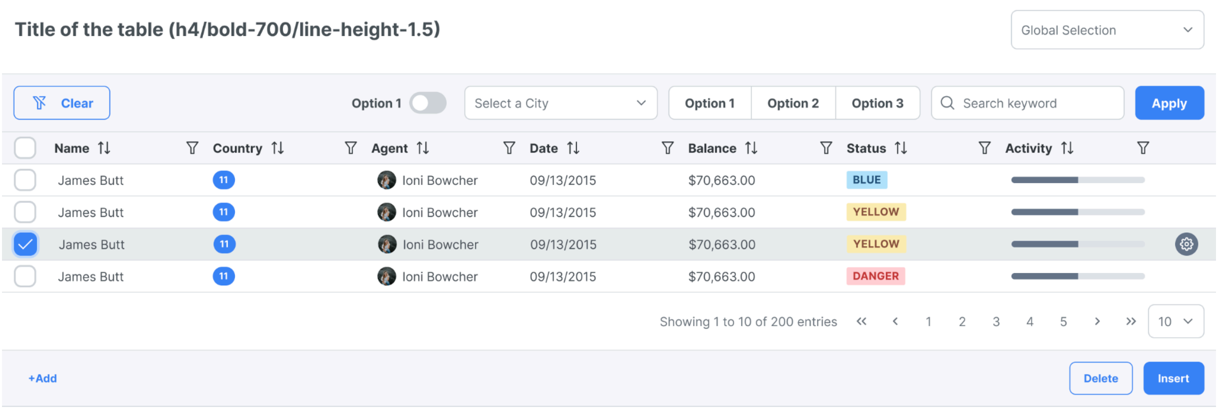
Options and Usage

Select v. copy methods

We are providing two methods for users to input data. Pros and cons are listed below for each of the methods.

Select → Insert

This option allows for inserting multiple rows of data at once. The downside is that controlling the location of where this data is inserted is less intuitive. The user must first enter data, then reorganize in the "inserted items" panel.



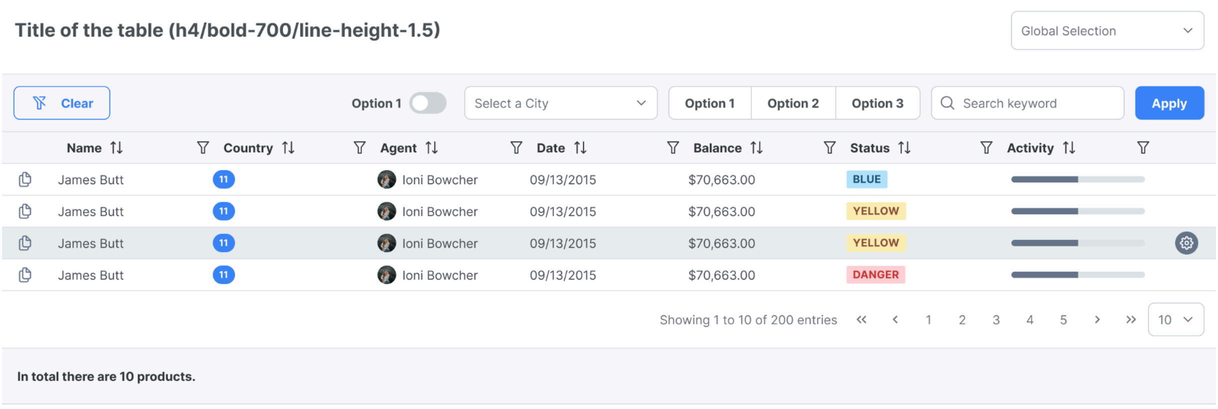
Title of the table (h4/bold-700/line-height-1.5)

	Name	Country	Agent	Date	Balance	Status	Activity
□	James Butt	11	Ioni Bowcher	09/13/2015	$70,663.00	BLUE	
□	James Butt	11	Ioni Bowcher	09/13/2015	$70,663.00	YELLOW	
☑	James Butt	11	Ioni Bowcher	09/13/2015	$70,663.00	YELLOW	
□	James Butt	11	Ioni Bowcher	09/13/2015	$70,663.00	DANGER	

Showing 1 to 10 of 200 entries

Copy → Insert

This option gives the user more control in where they can insert content. The downside is that it limits the user to inserting only one row of content at a time.



Title of the table (h4/bold-700/line-height-1.5)

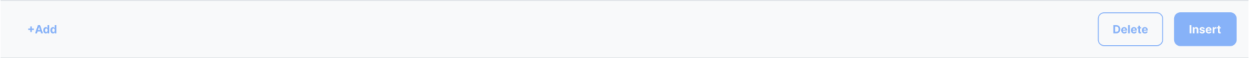
Name	Country	Agent	Date	Balance	Status	Activity
James Butt	11	Ioni Bowcher	09/13/2015	$70,663.00	BLUE	
James Butt	11	Ioni Bowcher	09/13/2015	$70,663.00	YELLOW	
James Butt	11	Ioni Bowcher	09/13/2015	$70,663.00	YELLOW	
James Butt	11	Ioni Bowcher	09/13/2015	$70,663.00	DANGER	

Showing 1 to 10 of 200 entries

In total there are 10 products.

Special considerations

If none of the checkboxes are checked, disable the bottom row of 'Add,' 'Delete,' and 'Insert' buttons (pictured here):



This space has been intentionally left blank.

Elements in this block

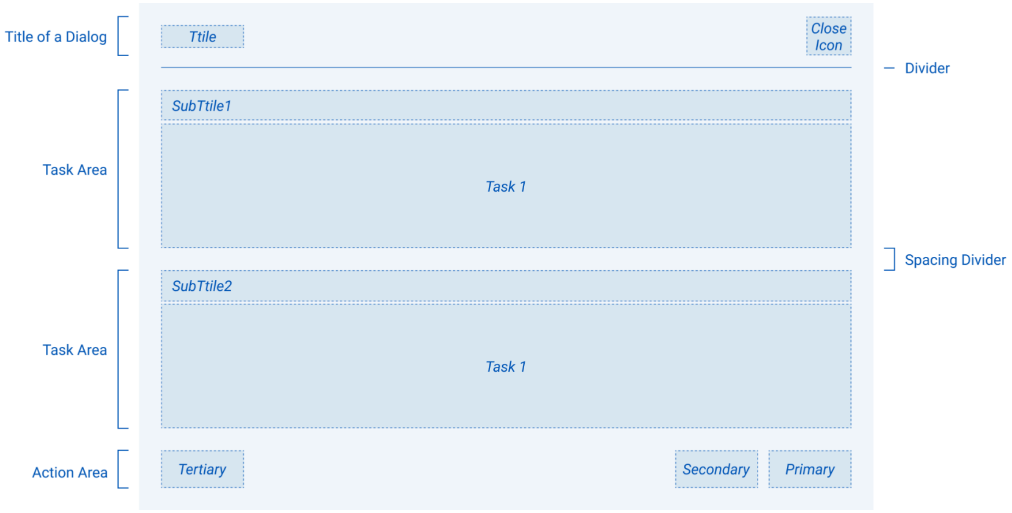
□ □ ☑ □	**Form > Checkbox** Use the ‘Checkbox’ component provided by PrimeNG.
✓ ✓	**Button > Button > Rounded Icon Buttons or** **Button > Button > Rounded Text Icon Buttons** Use the ‘Button’ component (both options listed above are fine) provided by PrimeNG.
⧉	**Copy Icon Button** Refer to the documentation for Icons.

Dialog

Anatomy

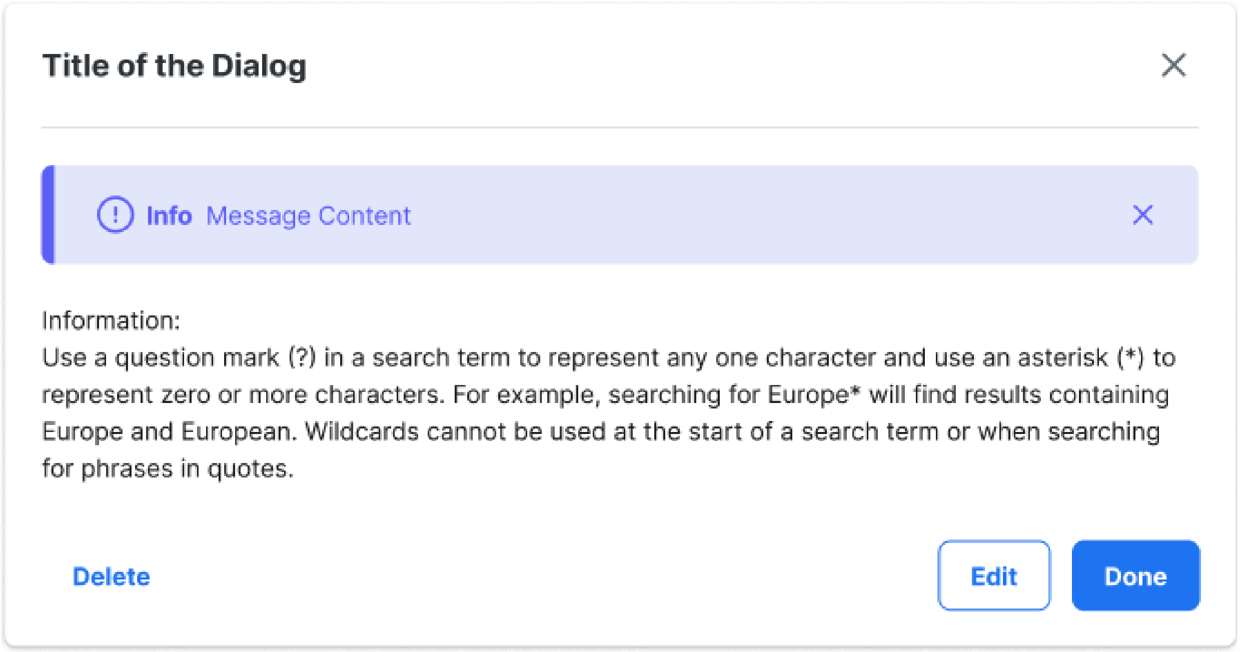
Template

Below is the template for the ‘Dialog’ custom block. Please note the terms, “Task Area” and “Action Area,” which will be referred to in the ‘Paddings’ section.



Making the block

Here is an example of a customized ‘Dialog’ block:



Information:
Use a question mark (?) in a search term to represent any one character and use an asterisk (*) to represent zero or more characters. For example, searching for Europe* will find results containing Europe and European. Wildcards cannot be used at the start of a search term or when searching for phrases in quotes.

Figma Inspect Link

See the “Dialog” page that you can find on the left panel.

Instructions

The instructions for making the customized dialog block is divided into two sections:

1.) **Block components:** the base component, buttons, and panels
2.) **Paddings:** the empty space around each component

Block components

1. Start with the 'ConfirmDialog - Basic' component provided by PrimeNG.

2. Add a divider under the header row ('Panel > Divider > Basic').
   Adjust the top and bottom paddings of the divider to 0px.

3. Replace buttons with the recommended buttons from the 'Components - Button' section. You will need to assign buttons as primary, secondary, and/or tertiary.

   a. By default, buttons are placed in the lower right corner. Keep buttons at this position.

   b. The primary button should be on the right side of the secondary button.

   c. Separate the primary and secondary buttons with a 14px gap.

   d. When adding a tertiary button, place it alongside the other buttons in a flexbox. Use the "space between" setting.

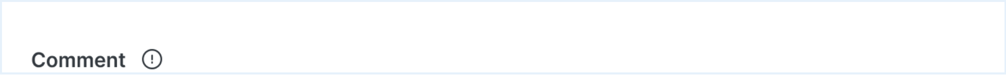
   e. Remove any buttons you do not need. In other words, do not implement a tertiary button for the sake of filling the empty space.

4. If your content warrants a subtitle, use the following: **[Figma Link]**

   a. Use subtitles if you need to group elements in the Task Area.

   b. Change text to "**h6/semibold-600/line-height-1.2**".

Example of the subtitle element:



Comment ⓘ

Paddings

In this section, we provide padding guidelines for three different situations:

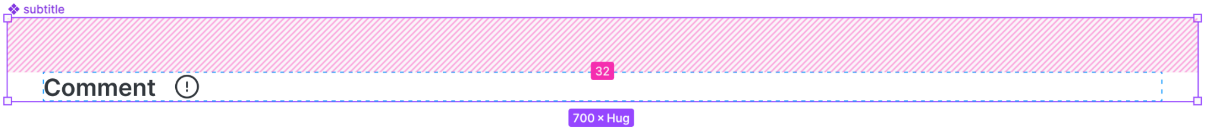
1.) Padding between Task Areas / Action Areas
2.) Padding between body elements, with subtitle
3.) Padding between body elements, without subtitle

In the images provided in this section, the padding is represented by the pink area with the diagonal lines.

Padding between Task Areas / Action Areas

The separation of contents the dialog box is primarily handled by increasing the size of the top paddings in the subtitle, Action Area, and Task Area elements:

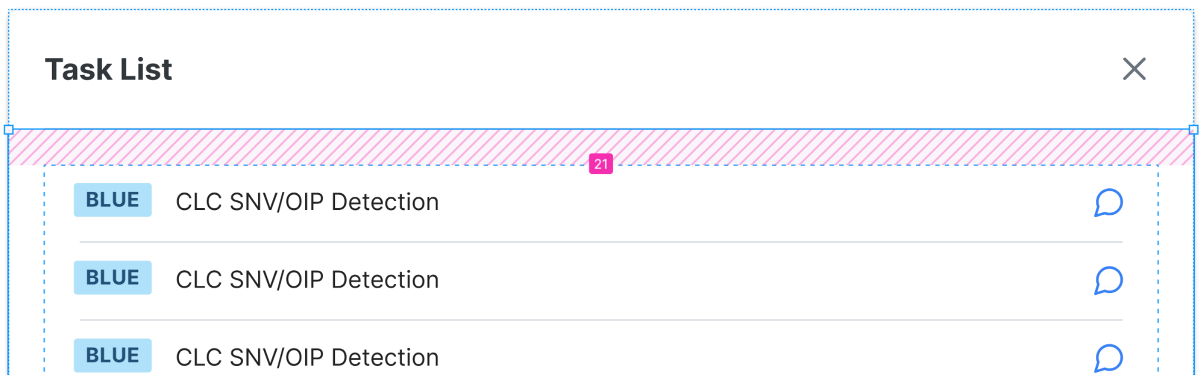
Padding top = 32px
Padding bottom = 0px



Padding between body elements, without subtitle

Padding top = 21px (e.g. the title, the divider, the message)
Padding bottom = 0px or 1px

- 0px will be the most common option
- 1px is reserved for elements that have an outside stroke (e.g. Text Area.)



Task List

BLUE CLC SNV/OIP Detection

BLUE CLC SNV/OIP Detection

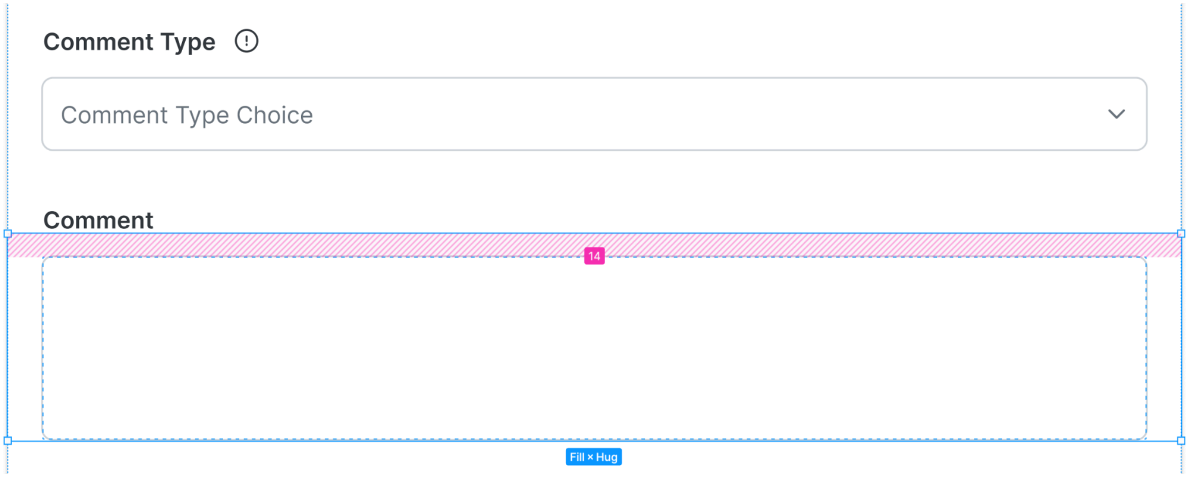
BLUE CLC SNV/OIP Detection

Padding between body elements, with subtitle

Padding top = 14px

Padding bottom = 0px or 1px

- 0px will be the most common option
- 1px is reserved for elements that have an outside stroke (e.g. Text Area.)



Comment Type

Comment Type Choice

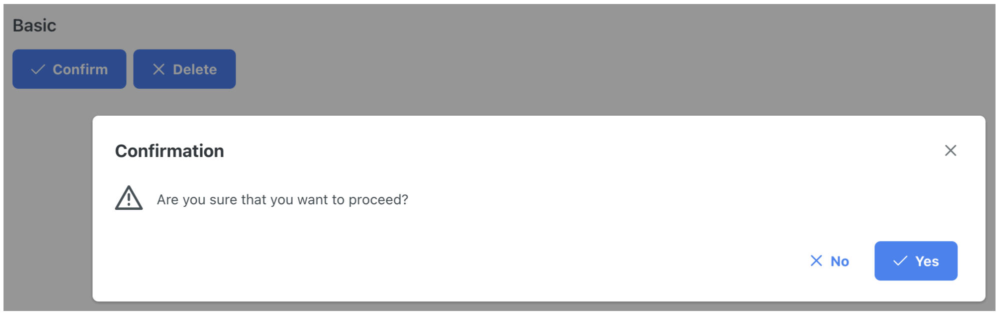
Comment

Options and Usage

Trigger Options

Overlay > ConfirmDialog > Basic

The trigger button should maintain an icon on the left. Buttons without an icon should represent for an individual action, while buttons with an icon should imply that there’s a following second step or an expanded content.

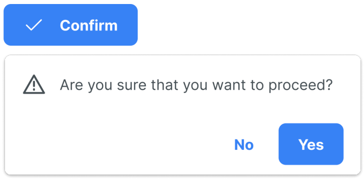

Basic

Confirmation

Are you sure that you want to proceed?

Overlay - ConfirmPopup

Please apply this pattern only to simple yes/no extensions.


Are you sure that you want to proceed?

Overlay - Dynamic Dialog

Please apply this pattern to the circumstance that the content in the dialog connects to a database. E.g. a table, a list.

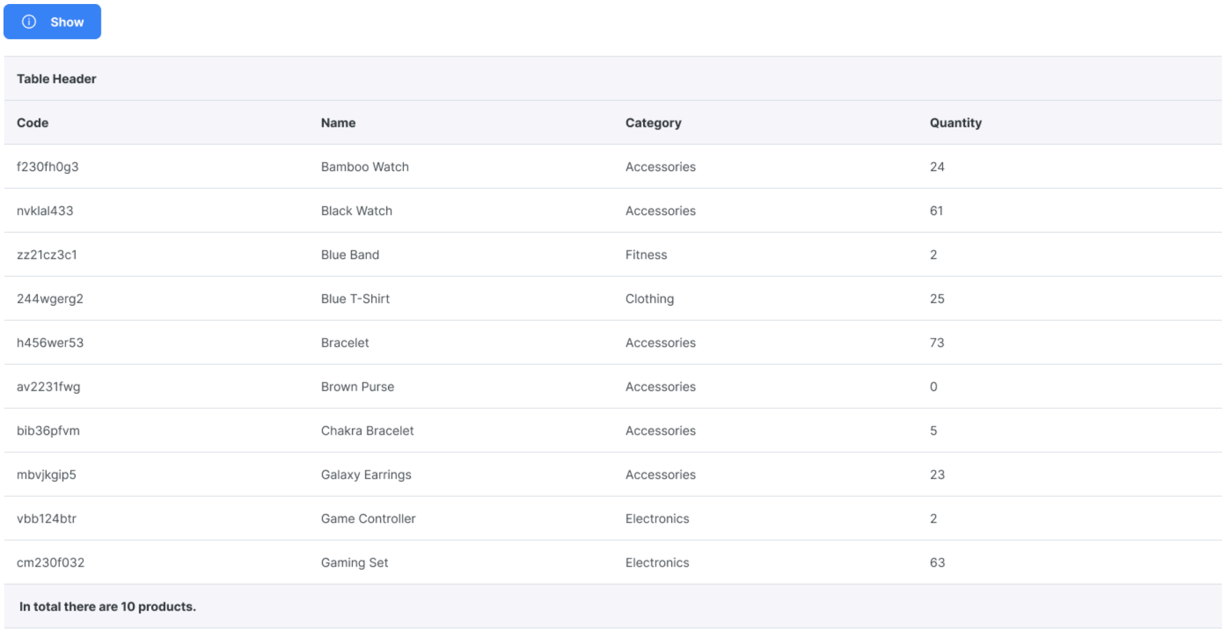


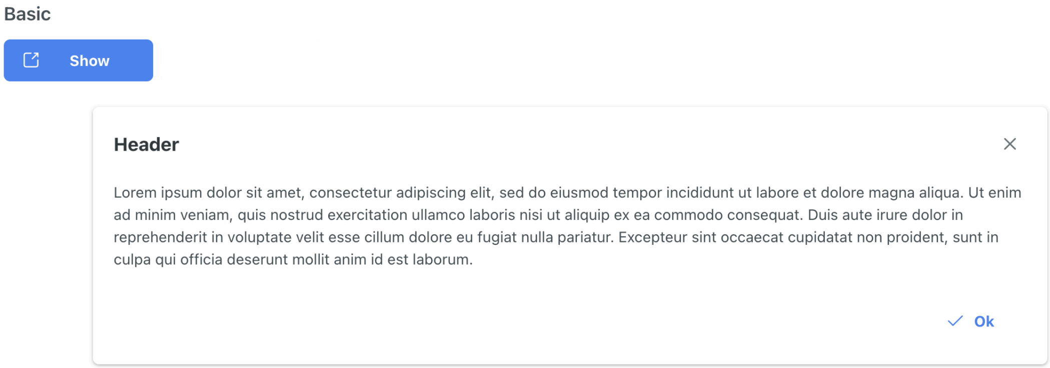
Table Header

Code	Name	Category	Quantity
f230fh0g3	Bamboo Watch	Accessories	24
nvklal433	Black Watch	Accessories	61
zz21cz3c1	Blue Band	Fitness	2
244wgerg2	Blue T-Shirt	Clothing	25
h456wer53	Bracelet	Accessories	73
av2231fwg	Brown Purse	Accessories	0
bib36pfvm	Chakra Bracelet	Accessories	5
mbvjkgip5	Galaxy Earrings	Accessories	23
vbb124btr	Game Controller	Electronics	2
cm230f032	Gaming Set	Electronics	63

In total there are 10 products.

Overlay - Dialog - Modal

The trait in this pattern that I want to emphasize is the background color underneath the white dialog panel. Please apply it to all dialogs that are more complex than a ConfirmPopup. E.g. all the examples below.



Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Overlay > Sidebar >

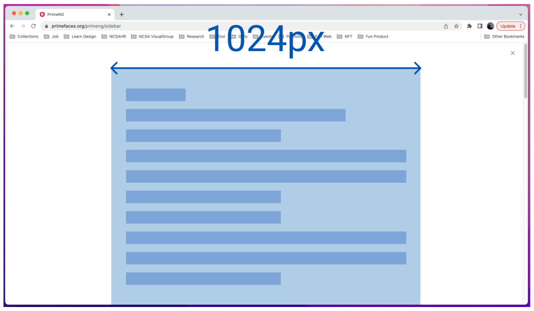
PrimeNG does not provide a name for this option. We will refer to this as the 'Full Screen Display' option.

'Full Screen Display' option opens the dialog box in full screen, and the user must click on the 'x' in the upper right corner in order to close it.

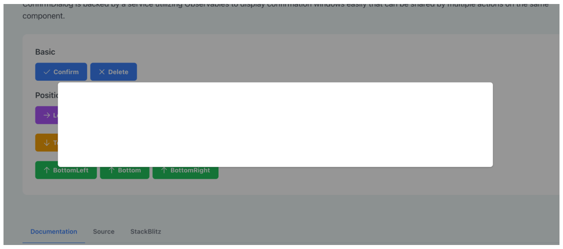
Use this option when the content does not fit in the standard dialog box.

When using this option, you will need to (see illustration below):

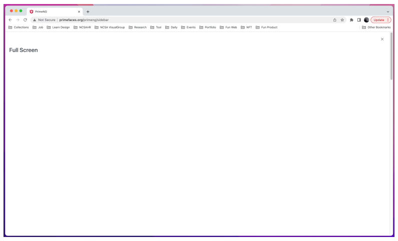
1.) Remove the underlying panel
2.) Transfer all elements into the 'Sidebar - Full Screen Display' component
3.) Set the width of the full screen display to 1024px
4.) Align the left and right margins of the content so that it is centered on the screen.
5.) Align the text to the left.



Below is an image of the dialog box (left) and the 'Full Screen Display' option (right).



→

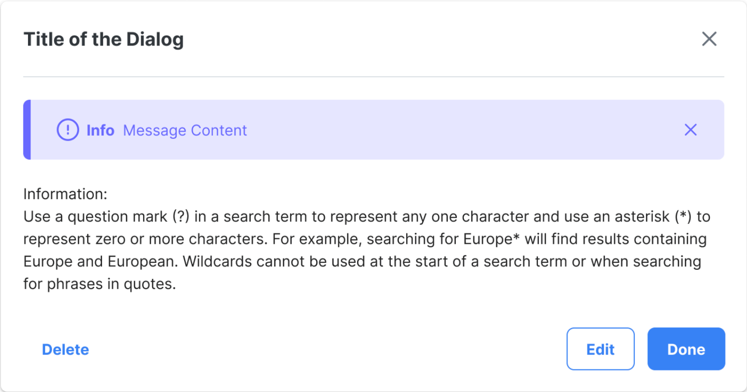


Element Positions

Maintain the sequence even if some items are taken off:

1. Message
2. SubTitle
3. Elements or a combination of elements chosen from the Component section above. Common elements are:
    a. Input
    b. Dropdown
    c. File Uploader

Examples



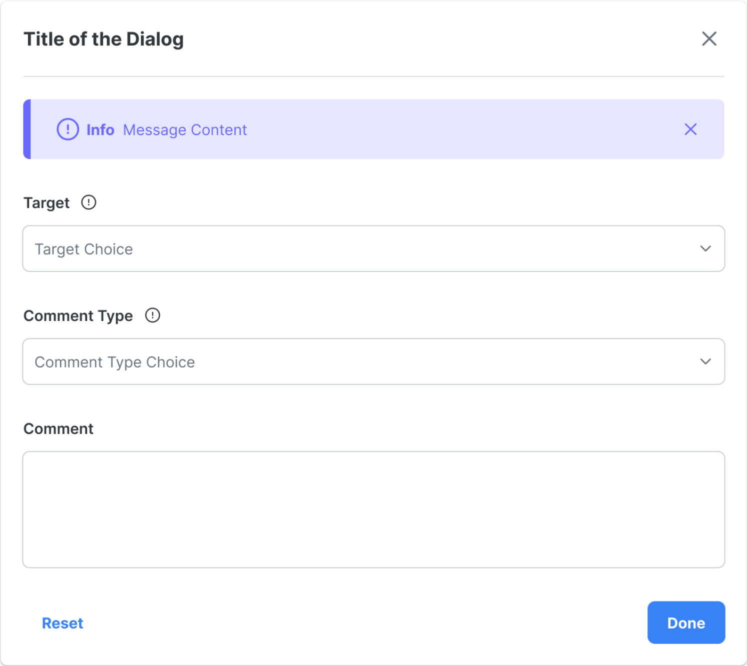
Title of the Dialog

Info Message Content

Information:
Use a question mark (?) in a search term to represent any one character and use an asterisk (*) to represent zero or more characters. For example, searching for Europe* will find results containing Europe and European. Wildcards cannot be used at the start of a search term or when searching for phrases in quotes.

Message

Plain text



Title of the Dialog

Info Message Content

Target

Target Choice

Comment Type

Comment Type Choice

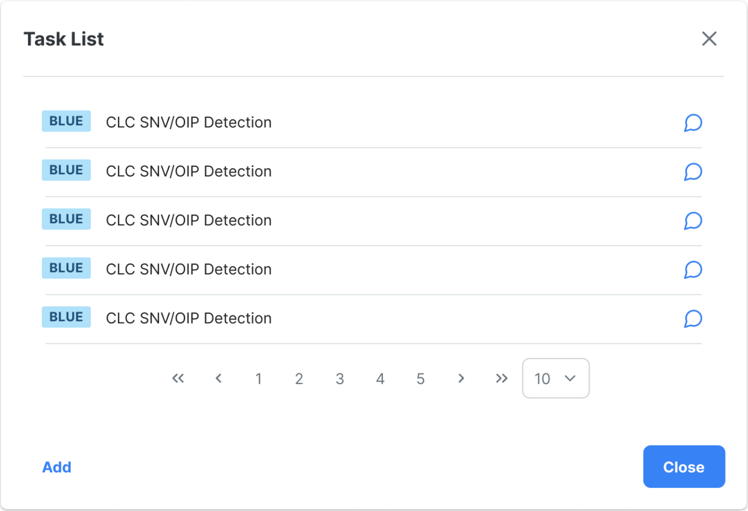
Comment

Message

SubTitle

Dropdown

Text Area



Task List

BLUE CLC SNV/OIP Detection

BLUE CLC SNV/OIP Detection

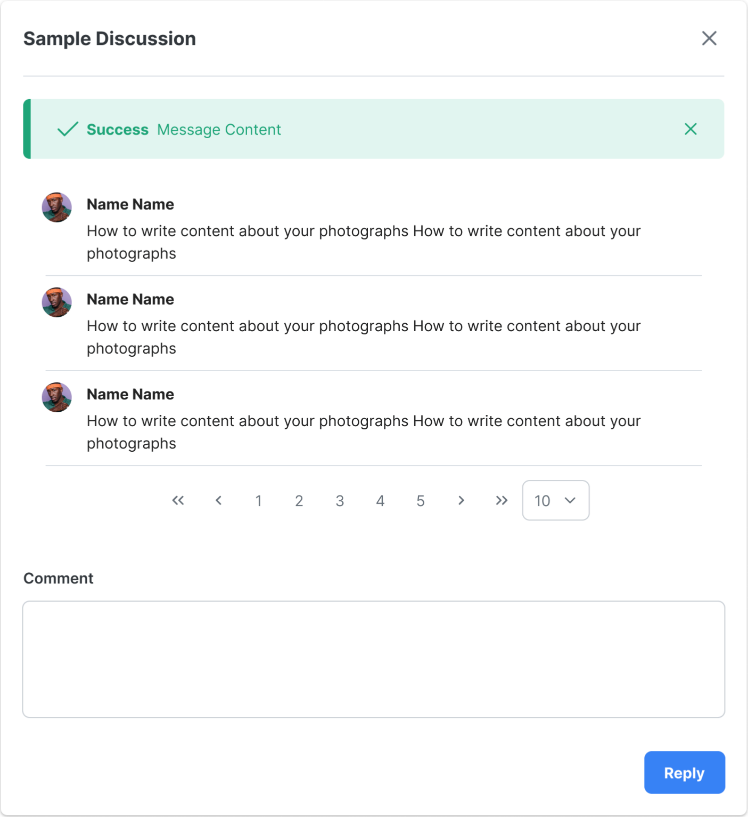
BLUE CLC SNV/OIP Detection

BLUE CLC SNV/OIP Detection

BLUE CLC SNV/OIP Detection

Customized list [Figma Link]

Paginator



Sample Discussion

✓ Success Message Content

Name Name
How to write content about your photographs How to write content about your photographs

Name Name
How to write content about your photographs How to write content about your photographs

Name Name
How to write content about your photographs How to write content about your photographs

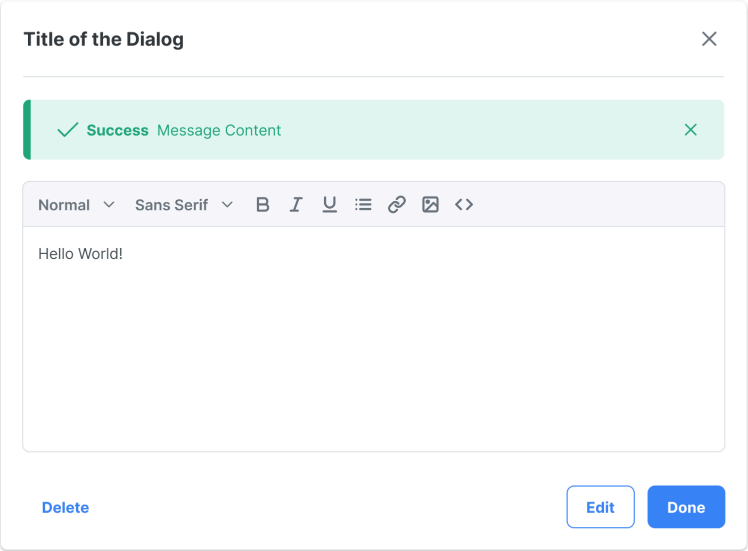
Message

Customized comment history [Figma Link]

Paginator

SubTitle

Text Area



Title of the Dialog

Success Message Content

Normal

Sans Serif

Hello World!

Message

Editor

Breadcrumb

Introduction

Definition

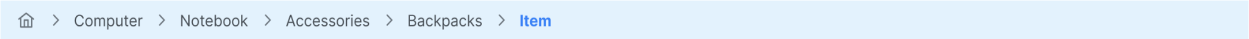
Breadcrumbs help users identify where they are in the application. This component is especially important in the context of GBS applications, since many pages look very similar to each other.

When to use

Given how similar pages are to one another, we recommend that Breadcrumbs be used **for every page** except for the home page. Adding Breadcrumbs to every page will significantly aid the user in navigation of the application.

Making the component

Here is an example of the customized breadcrumb:



Figma Inspect Link

See the “Breadcrumbs” page that you can find on the left panel.

Instructions

1. Start with the ‘Breadcrumb’ component provided by PrimeNG
2. Remove the stroke
3. Set the round corner to 0px
4. Set the background color to “`—primary-lightest-color`”
5. Change all text to “**h6/regular-400/line-height-1.5**”
6. Change the top and bottom padding to 7px (the whole height of the Breadcrumb component should be 35px)
7. Change the text color of the breadcrumb’s last item to the primary color. In the example above, the text color for ‘Item’ should the primary color.

Small Paginator

Introduction

Definition

Pagination is a way to display content in a paged format.

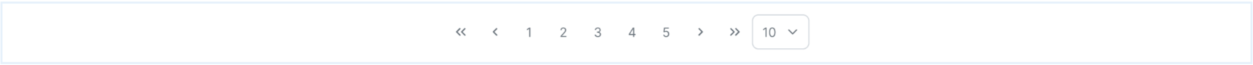
When to use

Generally, we recommend pagination over scrolling. Pagination works best when users are looking for specific pieces of content AND they have a general sense of count / where the information is positioned within the data. Since the data displayed in GBS applications are generally ordered (by date, alphabetical, etc.), users usually have a general sense of where to find their data.

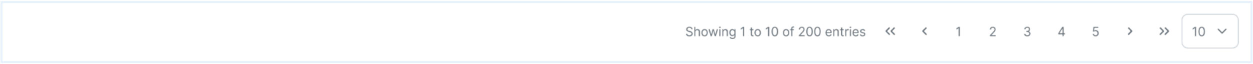
Making the component

Below are examples of the customized paginator. In the images below, please note that the light blue stroke is NOT part of the custom component. We have included it here to show the boundary of the paginator component.

Small Paginator - A



Small Paginator - B



Showing 1 to 10 of 200 entries

Figma Inspect Link

See the "Small Paginator" page that you can find on the left panel.

Instructions

In order for paginators to fit into the limited and condensed space of the GBS applications, we are decreasing the size of the paginator.

5. Start with the 'Paginator' component provided by PrimeNG
6. Change each button cell from 42*42px to 36*36px
7. Change the height of the dropdown selection to 36px.

Note: Keep all other traits as provided by PrimeNG (e.g. width of dropdown, typographies, icons)

Options and Usage

Small Paginator - A

Use the 'Small Paginator - A' when users do not need to know the total number of data points available (e.g., dialogs).

Small Paginator - B

Use the 'Small Paginator - B' option when users might need to know the total number of data points available. This is a vital piece of information for users, since it gives them a relative framework for where their data may exist. This greatly aids in navigation of data, especially in cases where users are using tables.