



PRODUCT UX & UI LIBRARY KIT V1.0

FINASTRA PATTERNS & STANDARDS



GETTING STARTED

This is a collection of symbols and text styles for rapid design and prototyping of layouts for the UXP & other web based Finastra products & platforms.

DOCUMENT STRUCTURE

THIS DOCUMENT HAS SEVERAL SKETCH PAGES AS FOLLOWS:

1.

Cover
2.

Getting Started: (this one), instruction on how to use the kit.
3.

Release Notes: Incremental release updates, what's new and issues descriptions.
4.

Style: The core visual styles of Finastra, including typography, color and more.
5.

Components: A collection of all assets that correspond to the Carbon Design System and Component Library.
6.

Templates: A collection of templates for usage or reference.
7.

Work In Progress: Components, patterns, standards and templates still under construction (minor versions)
8.

Deprecated: A collection of items superseded or no longer considered efficient for use.
- Symbols

SYMBOLS - USAGE

Use Insert > Symbols to add symbols from the Finastra Product UX & UI Design Kit symbol library to your document. They are organized in groups including components, icons and add-ons. Use these symbols and templates as a base for your design or as a drag-and-drop design system.

TEXT STYLES - USAGE

Apply text styles to add pre-determined text styles from the Finastra Product UX & UI Design Kit text style library to your document. Keep in mind that text styles are kept in sync, so the style changes you make to one text layer are propagated to all layers using that style. Text styles are based on the typography that is described on the styles page of this document.

DOCUMENTATION STYLE

The documentation throughout the kit has the following structure:

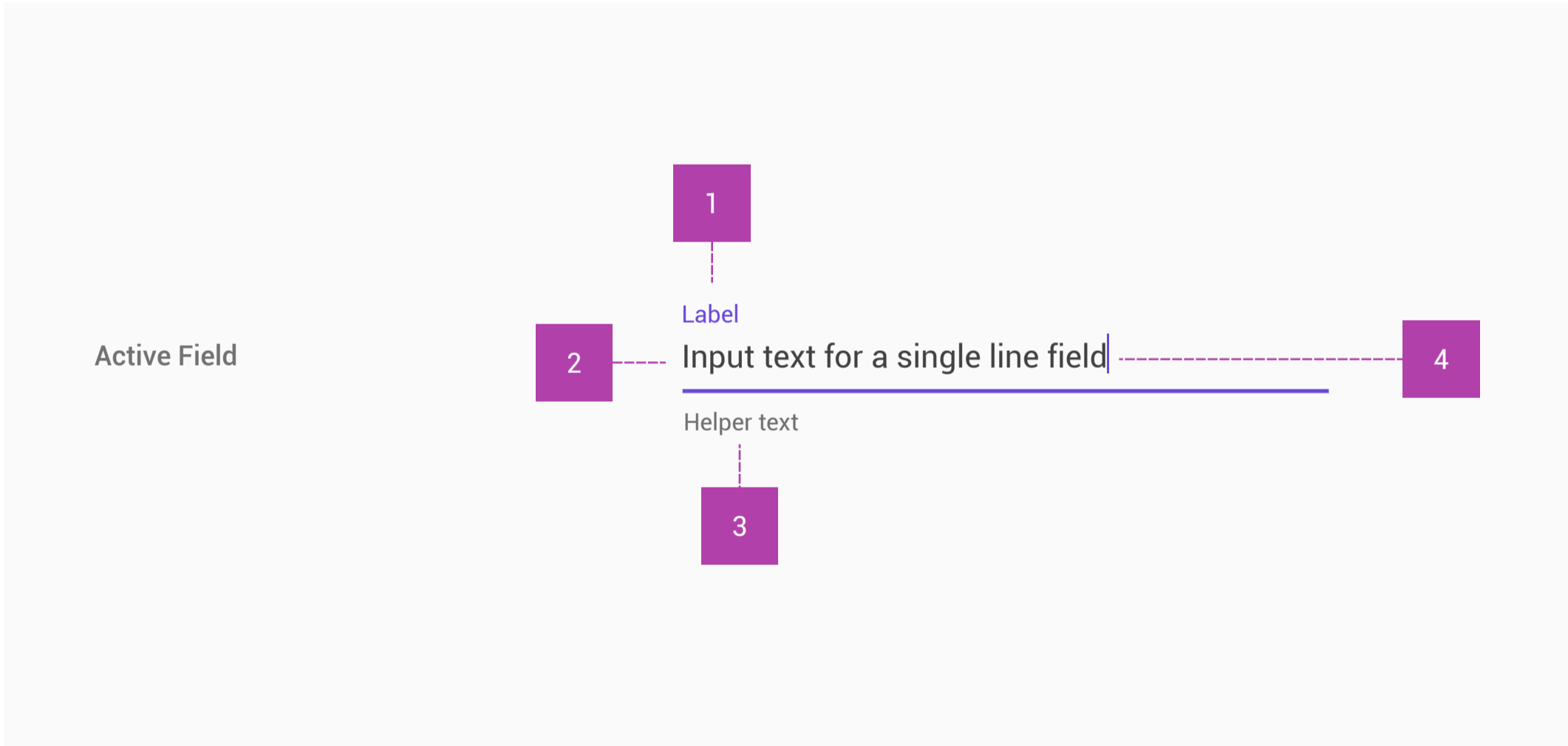
TITLE

HEADING 1

Heading 2

Cras quis nulla commodo, aliquam lectus sed, blandit augue. Cras ullamcorper bibendum bibendum. Duis tincidunt urna non pretium porta. Donec id dignissim nunc. Donec elit ante, eleifend a dolor et, venenatis facilisis dolor. In feugiat orci odio, sed lacinia sem elementum quis.

Nam condimentum vitae ligula vel ornare. Phasellus at semper turpis. Nunc eu tellus tortor. Etiam at condimentum nisl, vitae sagittis orci. Donec id dignissim nunc. Donec elit ante, eleifend a dolor et, venenatis facilisis dolor. In feugiat orci odio, sed lacinia sem elementum quis. Aliquam consectetur, eros et vulputate euismod, nunc leo tempor lacus, ac rhoncus neque eros nec lacus. Cras lobortis molestie faucibus.



State	Bottom Line	Label Text	Input/Placeholder Text
Idle & Empty	Thickness: 1px Color: C50	Type: Roboto 16px Color: C75 Spacing: 16px top padding, 8px bottom padding, inset with bottom line	
Roboto Medium	Roboto Medium: Roboto Light Color: C50		

TITLE

ATOM

TITLE

MOLECULE

TITLE

ORGANISM

TITLE

TEMPLATE

TITLE

PAGE

TITLE

STYLE



RELEASE NOTES

Sketch Libraries

The Finastra Product UX & UI Library Kit now includes Sketch Libraries, a better way to share and version symbols across Sketch files.

Package Contents

The package for the Kit now includes the required fonts, the document colour palette and any necessary sketch plugins to assist with setting up the file for any prototyping need.

New in Components include:

- Input Fields
- Dropdowns
- Date Picker
- Tooltips
- Selections
 - Checkboxes
 - Radio Buttons
 - Switch Toggle
- Buttons
- Hyperlinks
- Icon Behaviour
- Steppers

New in Templates include:

- Global Header/Navigation
- Web Based Login Screen

New in WIP include:

- Tabs
- Search (+field autocomplete)
- System Notifications
- Toast Notifications
- Modal Boxes
- Grid System

Skecth Version

The Sketch version used is 49.3 (51167) and there are not any know issues on the usage of this kit.

COLOR PALETTE

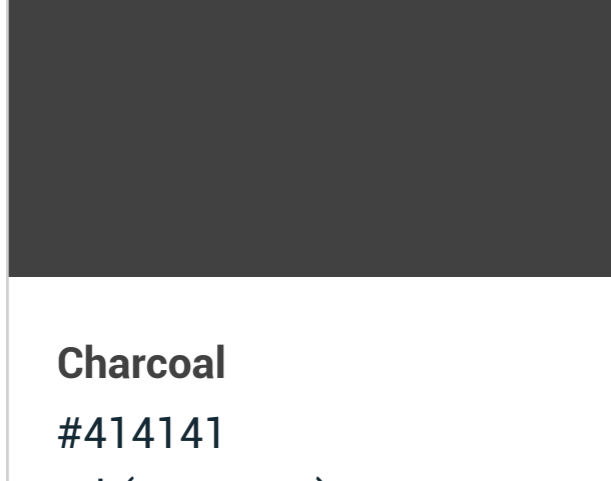
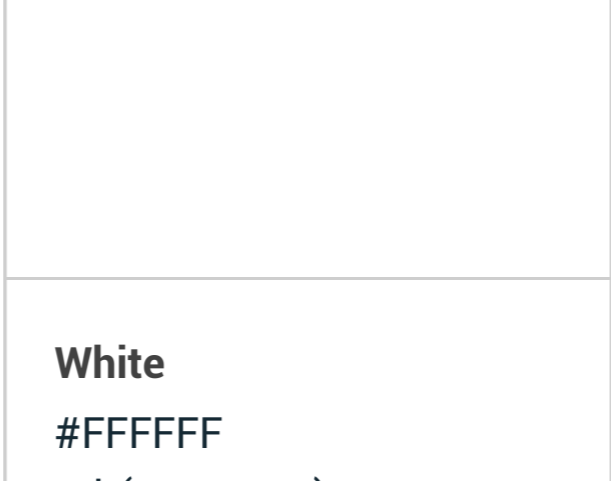
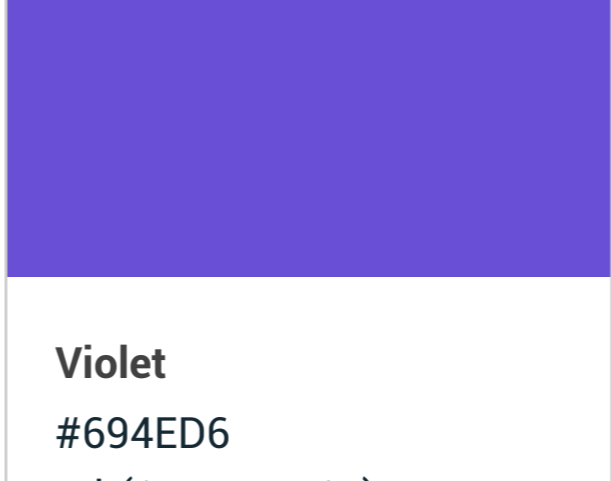
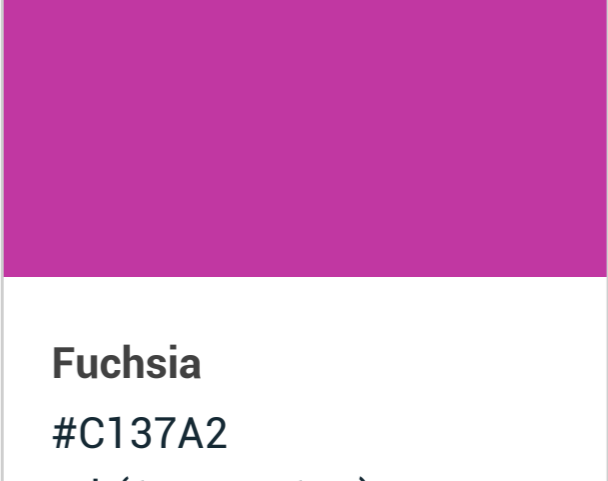
 **STYLE**

Our color palette plays a vital part in how we present ourselves. Our clean canvas of White and Charcoal represents our open approach, whilst a fusion of vibrant colors bring energy and differentiation.

Fuchsia and Violet are our lead colors, helping us to be unique and stand apart from our competitors. (Brand Guidelines v3.0)

PRIMARY COLOR PALETTE

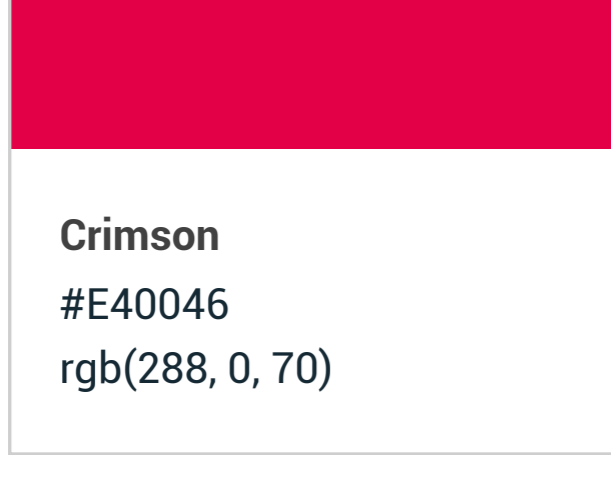
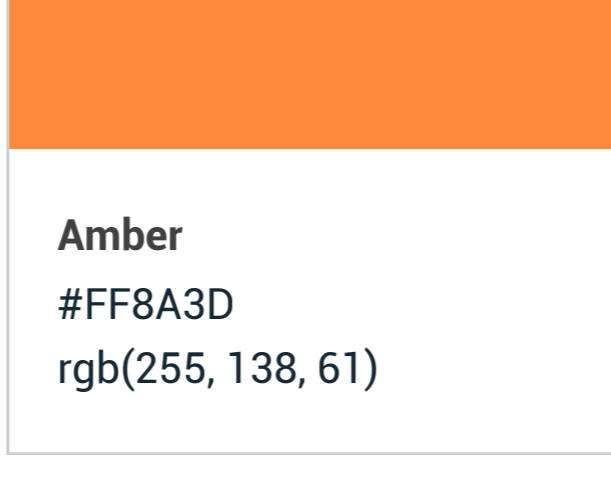
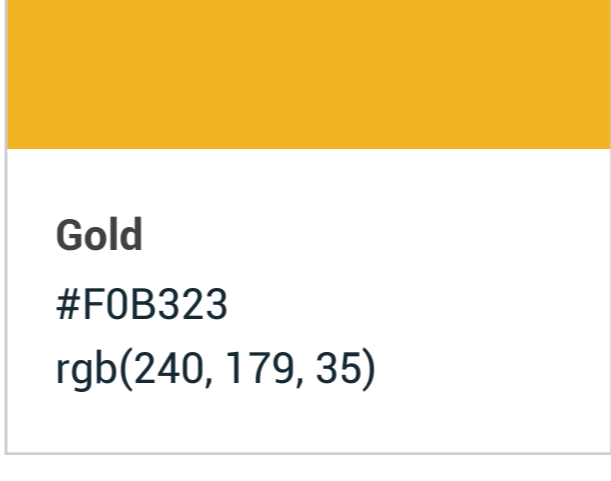
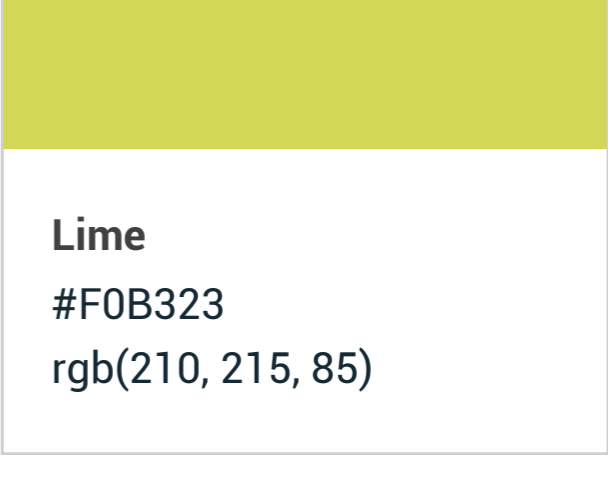
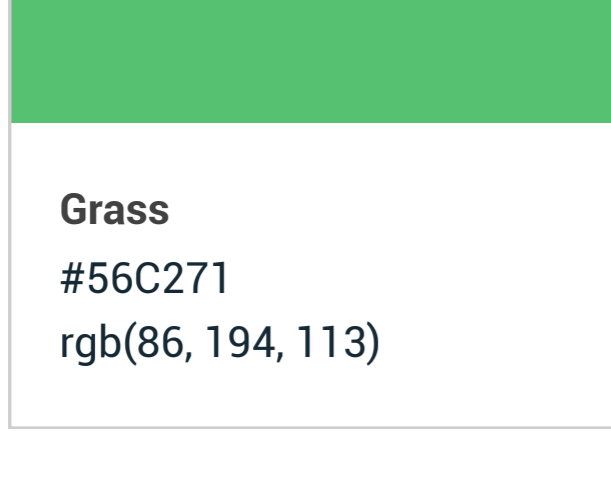
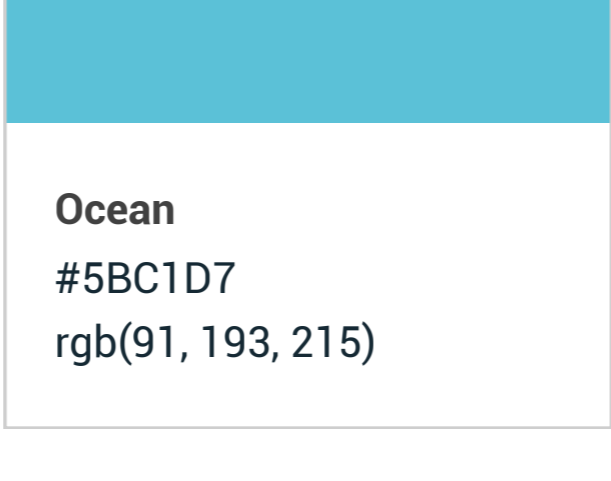
Our primary color palette consists of White, Charcoal, Violet and Fuchsia. These colors are present across most touch points from marketing to product.

			
Charcoal #414141 rgb(65, 65, 65)	White #FFFFFF rgb(65, 65, 65)	Violet #694ED6 rgb(105, 78, 214)	Fuchsia #C137A2 rgb(193, 55, 162)

EXTENDED COLOR PALETTE

Our extended palette works alongside Fuchsia and Violet to complete a full fusion of colors across the spectrum.

Secondary colors must always be used with Fuchsia and Violet. They must never replace our primary colors.

			
Crimson #E40046 rgb(288, 0, 70)	Amber #FF8A3D rgb(255, 138, 61)	Gold #F0B323 rgb(240, 179, 35)	Lime #F0B323 rgb(210, 215, 85)
			
Grass #56C271 rgb(86, 194, 113)	Ocean #5BC1D7 rgb(91, 193, 215)		

COLOR USAGE








Color brings a design to life. Color is versatile; it's used to express emotion and tone, as well as place emphasis and create associations. Color should always be used in meaningful and intentional ways in order to create patterns and visual cues. (Carbon)

COLOR ROLES





A color role refers to the systematic usage/s of a color value. (Carbon)

Each colour name is a combination of the first letter of the associated colour name and the tint perecentage. So C75 is Charcoal with a 75% tint.





Charcoal & White

	C100	#414141	Primary text, body copy Secondary text, subtle text
	C75	#717171	Hint text, emphasis border Subtle border, tertiary background
	C50	#A0A0A0	Secondary border, default background, field background, separators Listing item hover color
	C25	#CFCFCF	Primary container background
	C10	#ECECEC	
	C05	#F5F5F5	
	W100	#FFFFFF	

Violet (primary brand color)

	V100	#694ED6	Interactive text, primary icon color, border highlight, emphasis background In volutpat ante semper diam molestie
	V75	#8F7BE1	Sed sit amet arcu aliquet Phasellus ligula ipsum, volutpat eget semper id, viverra eget nibh
	V50	#B4A6EA	
	V25	#D9D3F5	

Fuchsia (supporting brand color)





	F100	#C137A2	Lorem ipsum dolor sit amet consectetur adipiscing elit
	F75	#D168BA	Ut pretium pretium tempor Ut eget imperdie
	F50	#E09BD0	
	F25	#EFCDE8	

USING THE EXTENDED (SECONDARY) PALETTE





The extended palette consists of all the useable shades of each color in the palette.

Tints of secondary colors are mainly reserved for use in infographics.





Crimson (red)

	R100	#E42346	Error highlight and error texts consectetur adipiscing elit
	R75	#EB5A75	Ut pretium pretium tempor Ut eget imperdie
	R50	#F191A2	
	R25	#F8C8D1	





Amber

	A100	#FF8A3E	Lorem ipsum dolor sit amet consectetur adipiscing elit
	A75	#FFA86E	Ut pretium pretium tempor Ut eget imperdie
	A50	#FFC49E	
	A25	#FFE2CE	





Gold (yellow)

	Y100	#F0B323	Warnings and highlighted messages consectetur adipiscing elit
	Y75	#F4C65A	Ut pretium pretium tempor Ut eget imperdie
	Y50	#F7D991	
	Y25	#FBECC8	





Lime

	L100	#D2D755	Lorem ipsum dolor sit amet consectetur adipiscing elit
	L75	#DEE180	Ut pretium pretium tempor Ut eget imperdie
	L50	#E8EBAA	
	L25	#F4F5D4	

Grass (green)

	G100	#56C271	Indication of success and the right outcome consectetur adipiscing elit
	G75	#81D295	Ut pretium pretium tempor Ut eget imperdie
	G50	#AAE0B8	
	G25	#D5F0DB	



Ocean (blue)

	B100	#5BC1D7	Lorem ipsum dolor sit amet consectetur adipiscing elit
	B75	#84D1E1	Ut pretium pretium tempor Ut eget imperdie
	B50	#ADE0EB	
	B25	#D6EFF5	

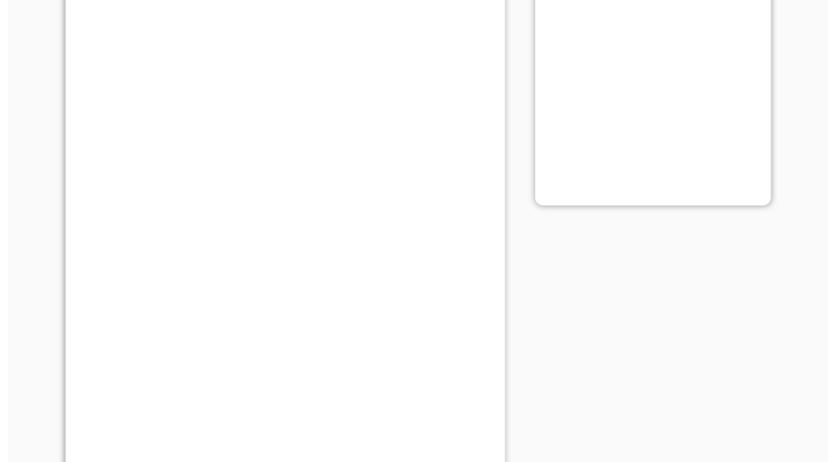
EXTENSIONAL COLORS

On top of our brand colors and their tints some extensional color support the visual structure of the interfaces.

Supportive color palette and styles

	BgDef	#FAFAFA	Default board background, base layer Dropshadow of items that use shadow for background highlight (e.g. cards, buttons)
	C50	#A0A0A0	box-shadow: 0px 0px 2px 0px #A0A0A0

Representation of layering with these colors



Our visual signature for interfaces is based on cards which create the containers for any information displayed on the screen.

OUR LOGO



OUR LOGO - SVG FORMAT



PRODUCT NAMES

A full list of our product names are available on our intranet in the document called Product Naming Hierachy. The names of the products are important to consider to be part of our interfaces as they serve as indentifiers to the given product used along with the product icons and our logo.

It is also important to understand the hierachy of naming for our products to consider placing and display of this element on our interfaces. Products along with their subproducts needs to be identified by the user.

Product names are used mainly on:

- the login screen
- product header

The lenght of the product names are also key to find the proper layouting for these elements.

Our longest product name is:

Fusion Global Liquidity and Risk Management

Our shortest product name is:

Fusion ECM

A majority of our products has independently called subproducts (modules) as well.

Examples:

Fusion Global PayPlus - SDK

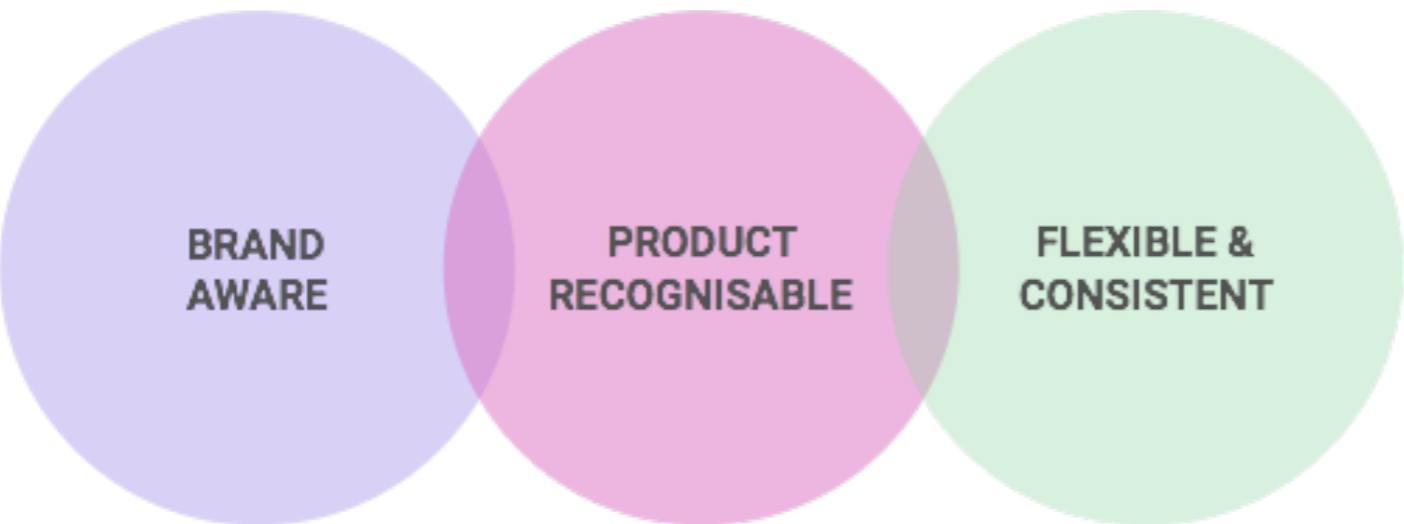
Fusion Sophis - Risk & Limits

PURPOSE OF THIS DOCUMENT

This document summarises all details to be able to comply to our product icon guidelines. Collects the requirements for elements to be presents, sizes and the application forms.

The guideline is mandatory to follow to satisfy the key principles of our logo system for products.

PRINCIPLES

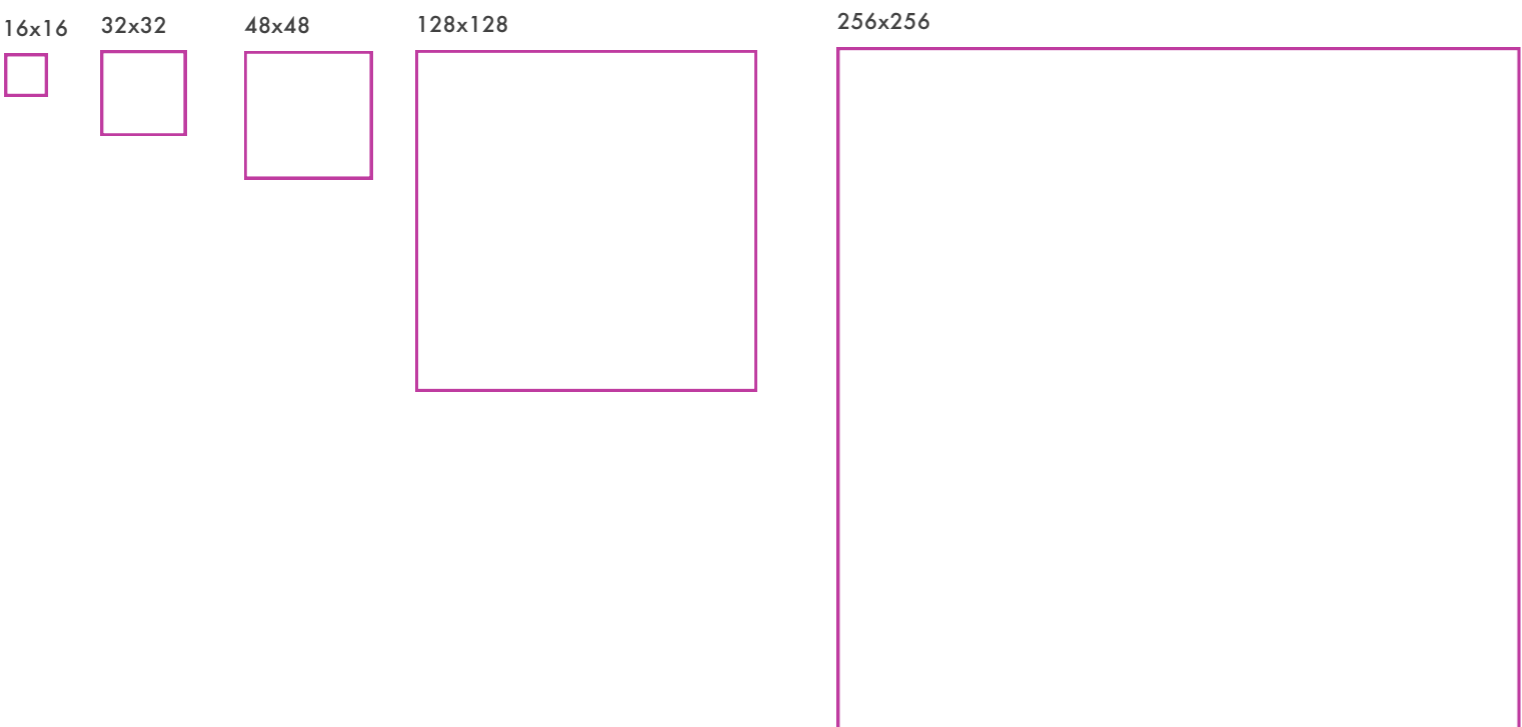


PLATFORM DEPENDENT REQUIREMENTS

Platform	Sizes and density	Places to apply (+ spec sizes)	Formats
Windows	Pixel based display. Wide range of sizes across different views starting from min. 16 px and goes up to 256 px. (sometimes even up to 512 px)	Tray: 16x16, 20x20 or 32x32 (depend on users' preferences) Desktop: 32x32, 64x64, 128x128 or 256x256 Application icon on window, in small size list items and quick system tray: 16x16	multi sized .ICO Should support transparent layers.
Mac OSX	Pixel based display. From @1 to @3 density exports to support retina display. This platform itself supports scaling of displayed elements. The icons should be scalable to unusual sizes as well. e.g. 41x41 px. Optionally can be vector graphic.	Dock: Scalable sizes. No default size. Finder listings: from 16x16 to 48 x 48 Application icon on window, notification centre: 16x16 Mac status bar: 24x24 Desktop icon: from 24x24 to 64x64	PNG format with alpha channels. (to support transparency) Optionally can be used as an SVG/EPS vector graphic icons. Should be generated as icon sets.
iOS & Android	Pixel based display. From @1 to @3 density exports to support retina display. This platform itself supports scaling of displayed elements. px. Optionally can be vector graphic. Might be truncated.	Home screen, menu app icon: 32x32 (if grouped then less than 16x16) Notification, notification centre: 16x16 App switch menu, listings: from 16x16 to 24x24	PNG format with alpha channels. (to support transparency) Optionally can be used as an SVG/EPS vector graphic icons. Should be generated as icon sets.
Web & miscellaneous	Pixel based display.	Browser tab favicon: 16x16 Notifications, notification centre: from 16x16 to 32x32 News letters, webpage, installation disc: No predefined size.	PNG and .ICO format with alpha channels. (to support transparency)

MINIMUM REQUIREMENTS FOR SIZES

By considering the platform requirements, Finastra's product icons should support the following sizes when talking about pixel based display.



STRUCTURE OF THE PRODUCT ICON SYSTEM

CONSIDERING BRAND ELEMENTS

Finastra Logo

Favicon

Main Brand Colors

Secondary Colors

Logo's Gradient

Optimised Logo Gradient

The Fusion Gradient

Optimised Fusion Gradient

Fusion Gradient's preferred direction for larger graphics

Typography

FUTURA EF - BOLD

Titles, main headings.
Not used as body font.
Used in Upper Case.

Futura EF - Medium

Subtitles, not used as body font.
Used in Upper Camel Case.

Roboto Regular

Body text, not used for headings. Can be used with several weights.

Our brand elements are used to create the basic structure for the product icons.
Considering the brand guidelines the following elements have been selected to be used in the icons:

1. Favicon: as this part represents our ribbon
2. White color: as the brand based on white
3. Fusion gradient: as it contains our brand colors, and the symbol of our products whereas it also creates a contrasted visual applied on top of white
4. Futura EF-Bold font: as it is our main brand font

PRODUCT NAMES

The icon should also include a clue to make the product identifiable. But in the same time this requirement should be satisfied by focusing on the principle that the icon should be flexible and easily generated.
This led us to the conclusion to use abbreviated names with letters in our product icons instead of iconography or different color schemas.
Important limitation: we use **maximum of 3 letters** to abbreviate a product.

Examples

FX Trade



FX


FXT

FxT

Don't

Do

Treasury



TCM

TRS

TRy

Don't

Do

Why?
Nobody will be able to find or create different icons for each product. Also iconography is hard to maintain can make the recognisability hard. Imagine if we have to Treasury products we use but the icon is the same on the Windows tray, however with letters a minor difference can be created between the two applications.

ANATOMY AND TYPES OF THE ICONS

Our products are structured in a way that we have master and sub products. By considering this portfolio structure we differentiate between these two in the icon system as well.

1. Default Public Icon: icon for the default Finastra Brand on public (ios & google playstore) app stores

2. Hero icons: icons for the master product e.g. TCM: Treasury and Capital Markets
These icons are structured to be prominent visual with bigger elements that apply the above mentioned clues.

3. Sub-product icons: icons for products that are sub-products of master products.
These icons remain a bit subtle comparing to the hero icons to convey the message that they are smaller / really specific applications.

DEFAULT ICON

Hero icons contain the following elements:

1. Base layer:
White rounder background with 10% rounded corners.

2. Finastra Logo:

HERO ICONS

Hero icons contain the following elements:

1. Base layer:
White rounder background with 10% rounded corners.

2. Product name / Text area:
Text area is using 40% of the height to create a space in the center for the letters.
Letters are 25% of the height so with bigger text area create a nice white space for the letters.
The font type is Future EF - Bold (our brand font) with the color of white #FFFFFF
As shown on the example above not just the icons with 3 letters follow these ratios. (TX above)

3. Ribbon, brand association: The ribbon is used with the Fusion gradient makes the icon aligned more to the brand with the color and its shape.

SUB-PRODUCT ICONS

Similarly to the hero icons sub-product icons contain the following elements:

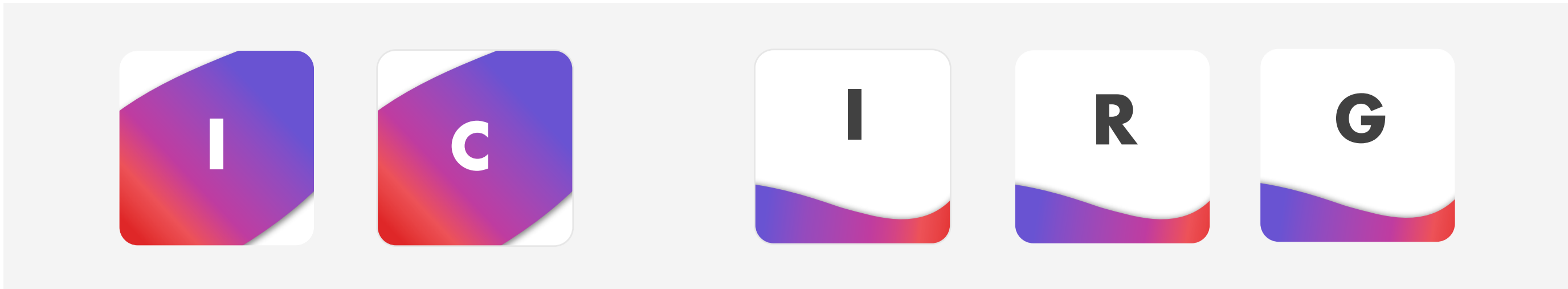
1. Base layer:
White rounder background with 10% rounded corners.

2. Product name / Text area:
Text area is using 40% of the height to create a space for the letters on the top.
Letters are 25% of the height so with bigger text area create a nice white space for the letters.
The font type is Future EF - Bold (our brand font) with the color of dark-grey #414141

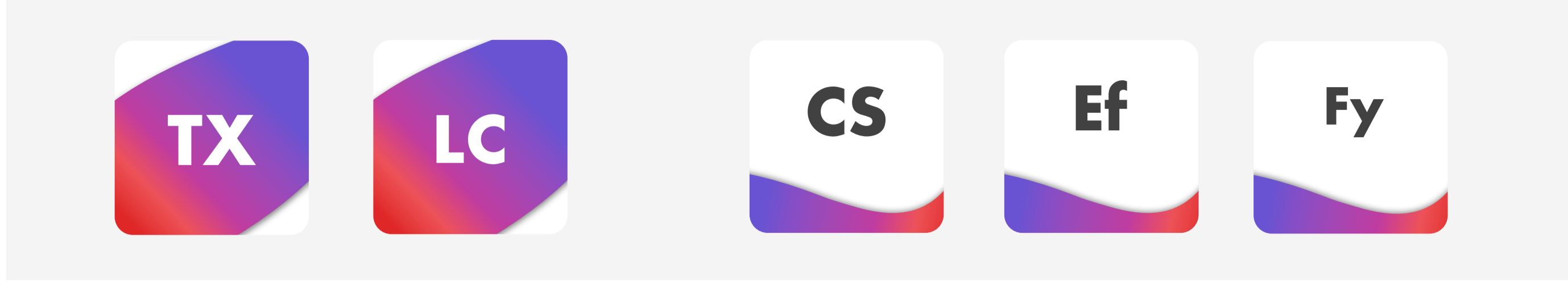
3. Ribbon, brand association: The ribbon is used with the Fusion gradient makes the icon aligned more to the brand with the color and its shape. On sub-product icons the gradient goes from purple to red.

DIFFERENT VARIATIONS OF THE ICONS

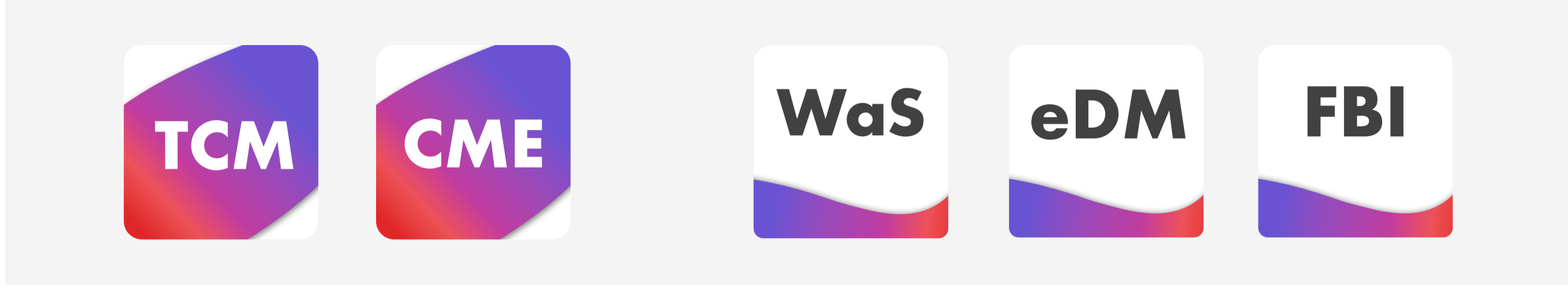
1 LETTER ICONS



2 LETTERS ICONS



3 LETTERS ICONS



APPLICATIONS

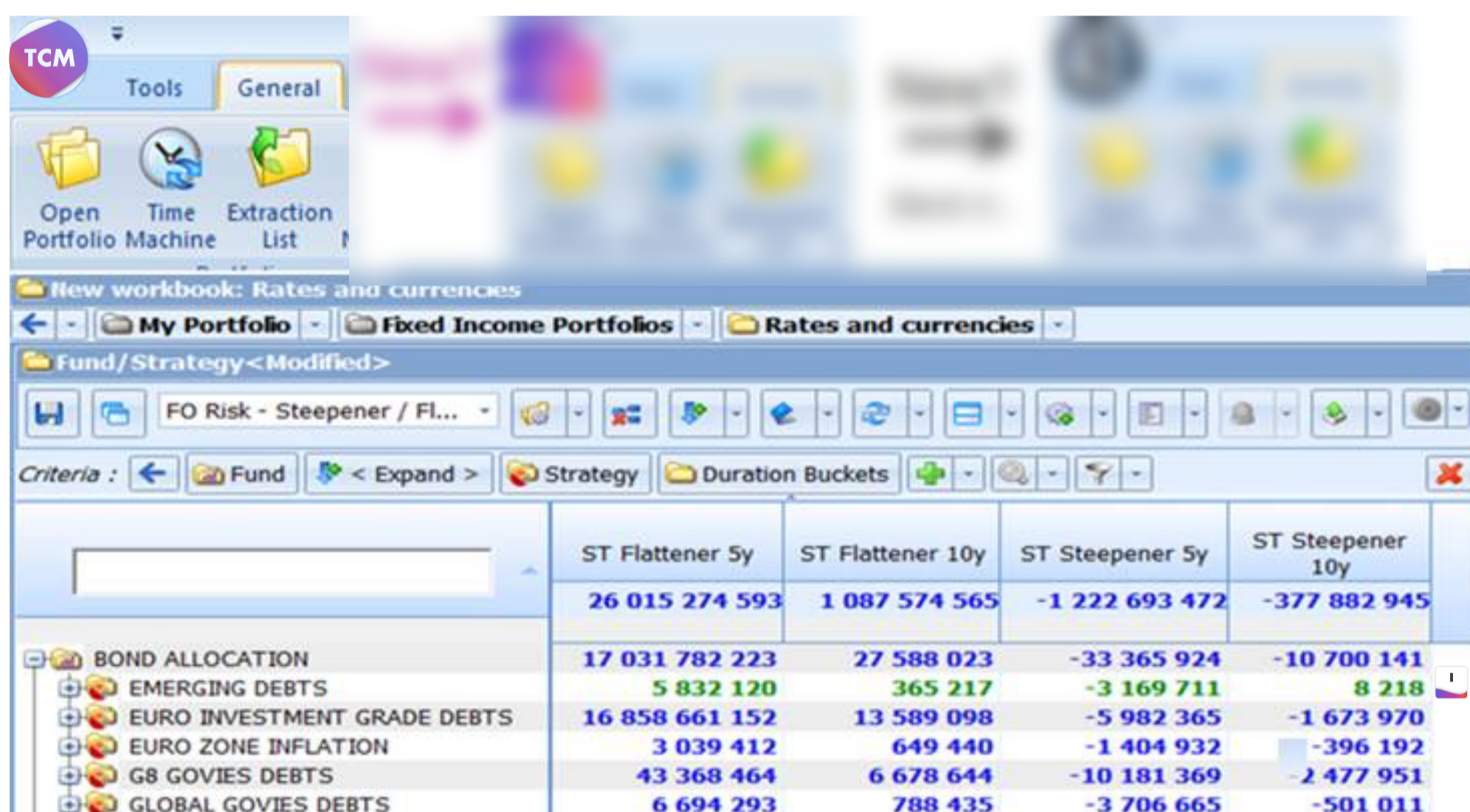
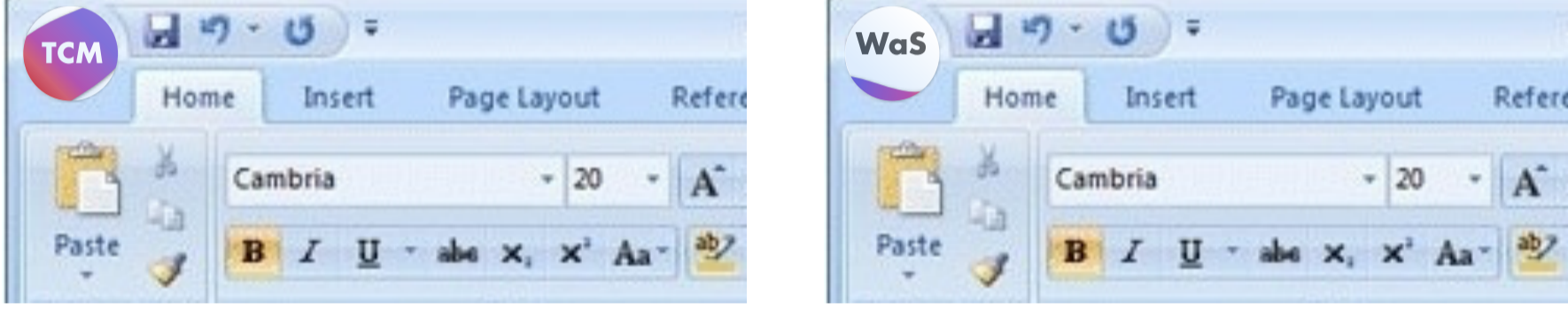
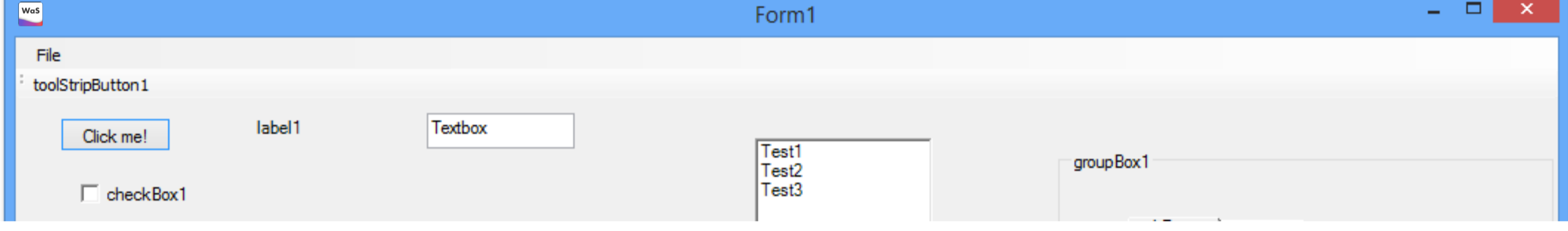
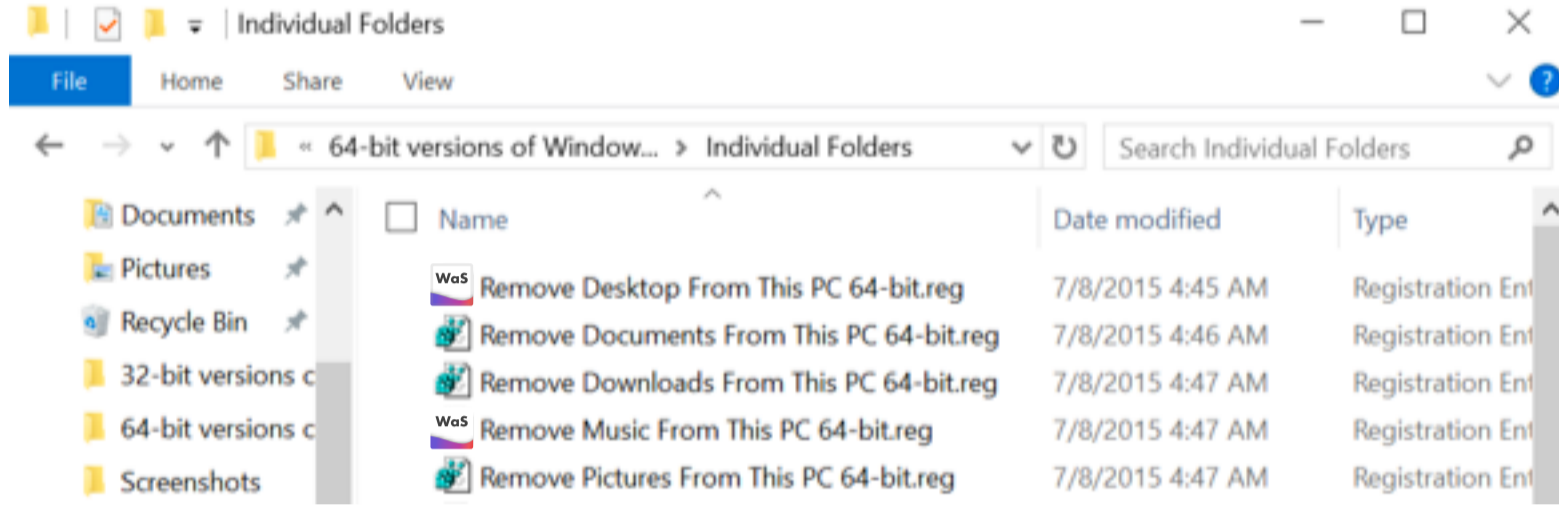
Different applications of the product icon shows the power of the letters and the ribbon as it makes the application prominent visually as well as the letters make them recognisable.

Windows

Sytem tray 32x32 and 16x16

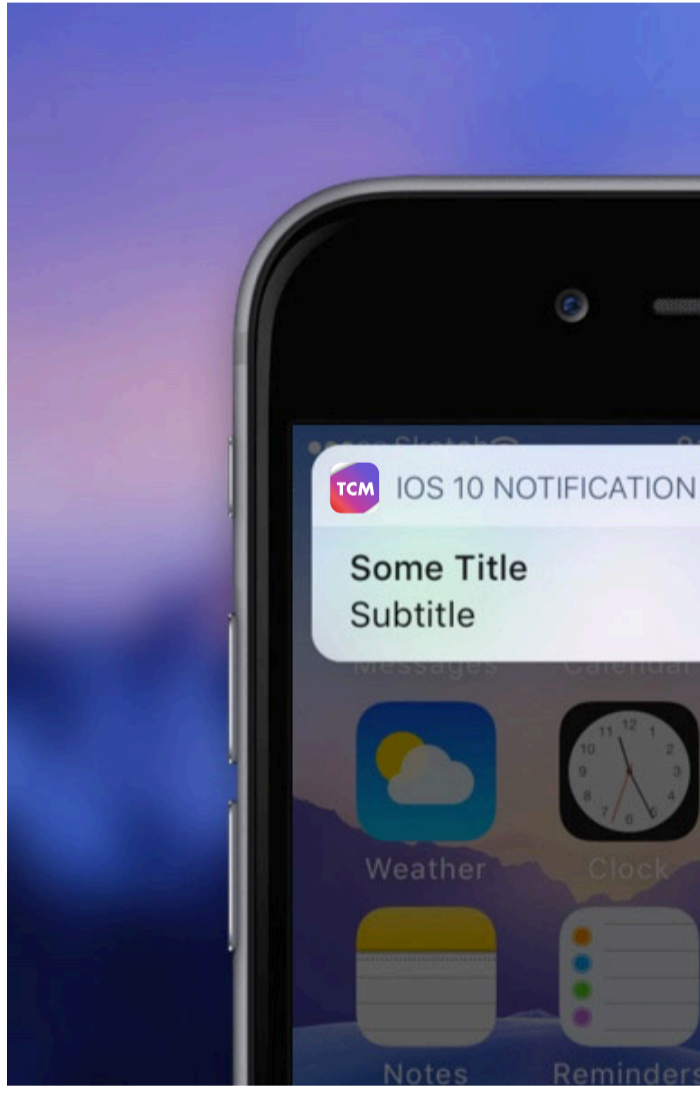
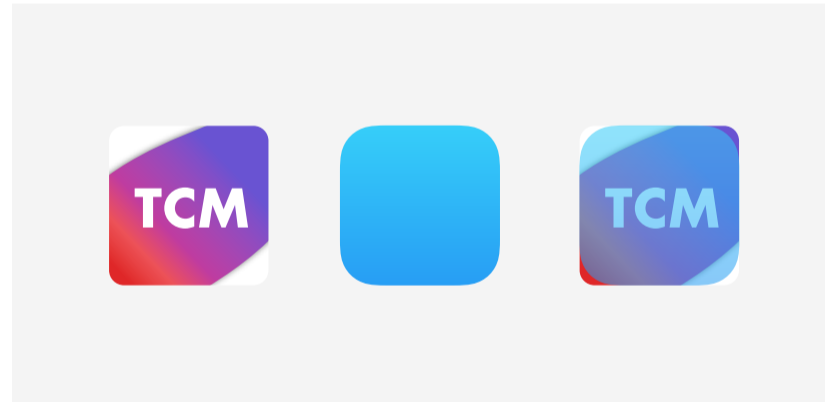


Listings and window title



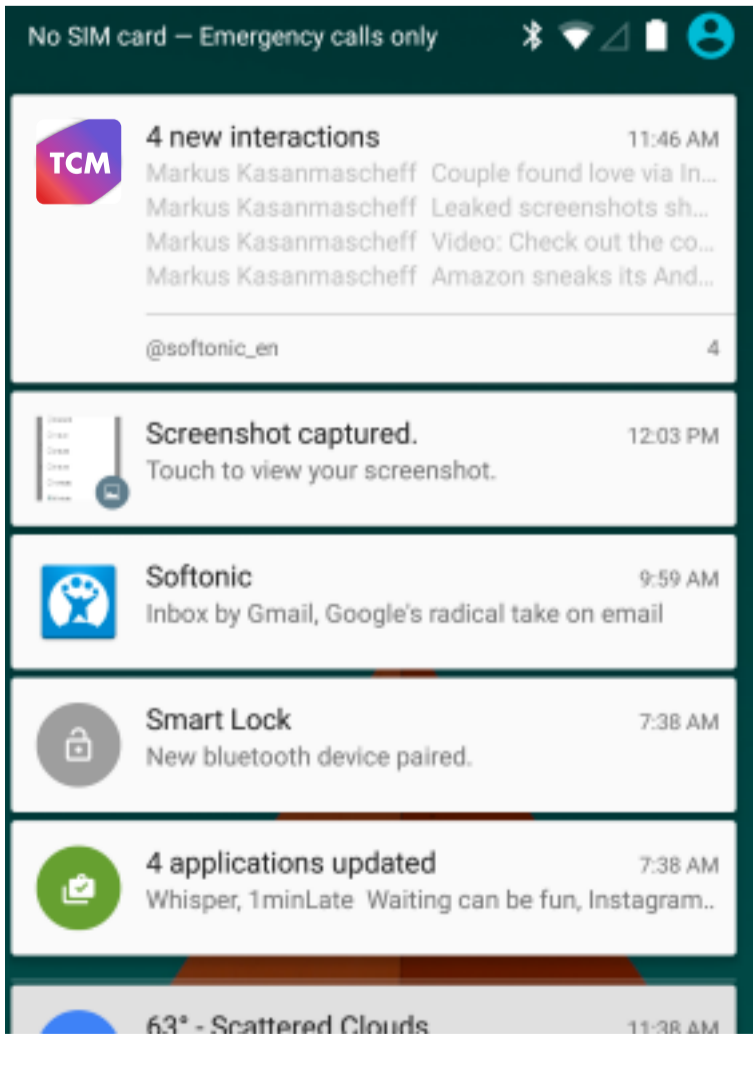
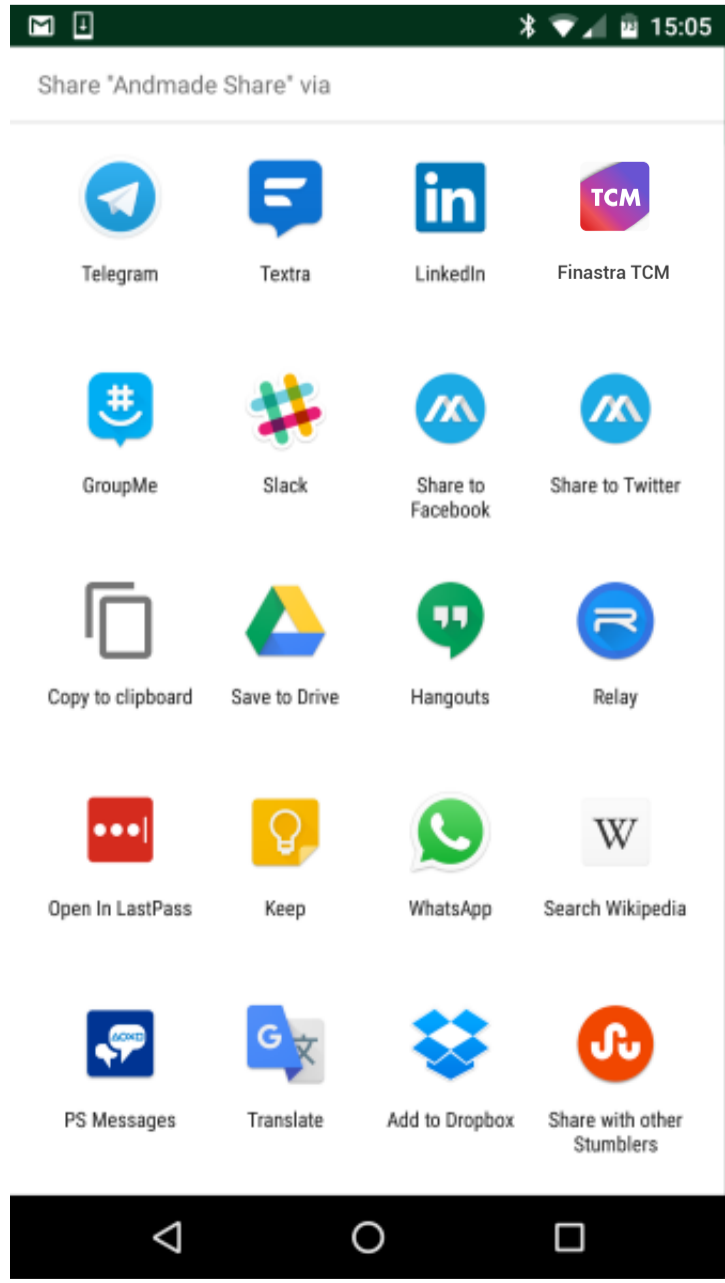
ios

Home screen and menu items and notifications

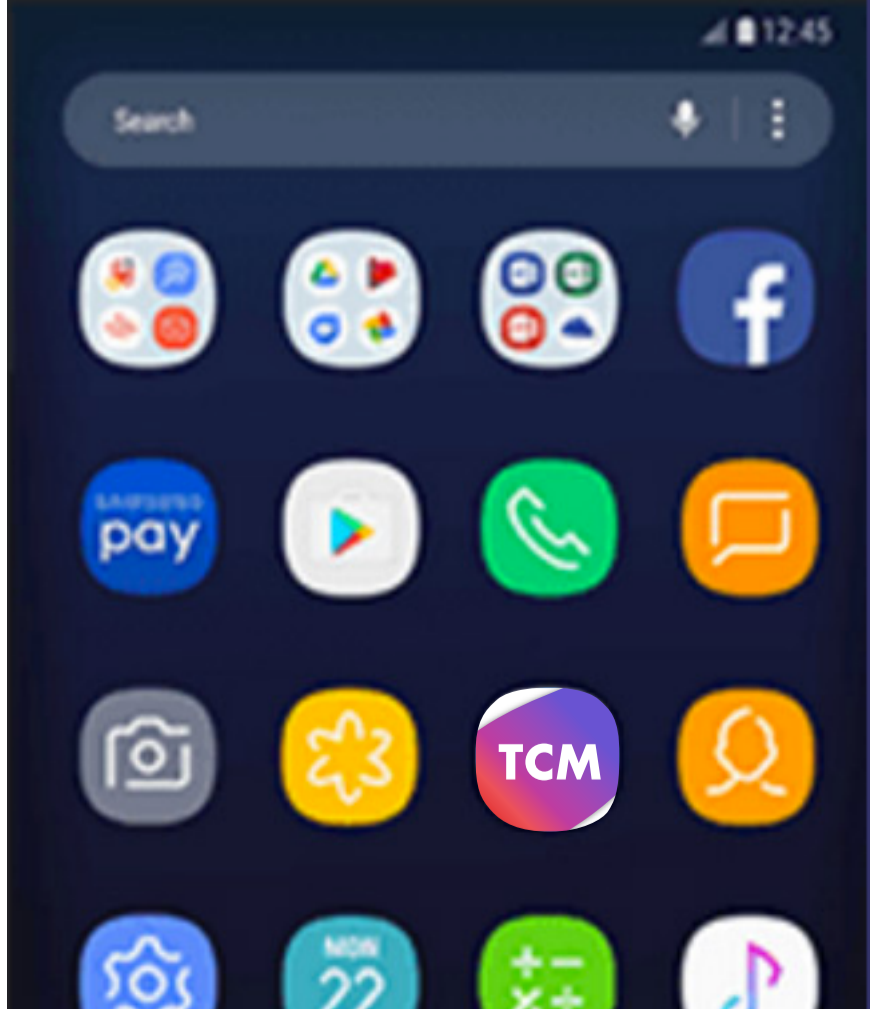
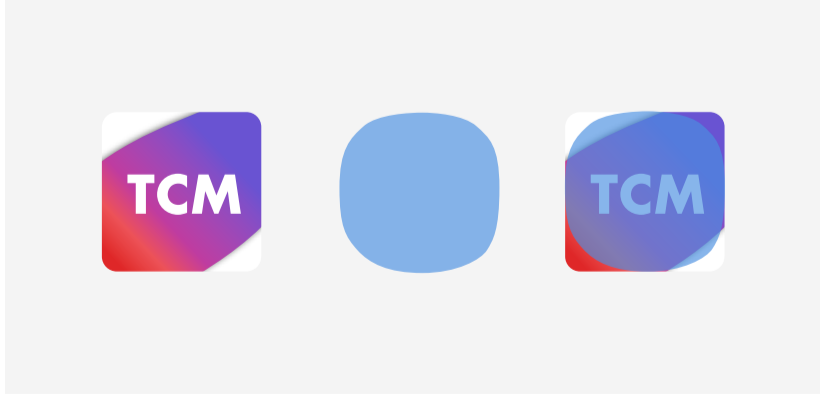


Android

Home screen and menu items and notifications



Samsung devices - Android M



INPUT FIELDS

Text inputs are used to allow people to input, edit and select text. Common input types include:

- Text: Usernames, descriptions, URLs.
- Numbers: Phone numbers, credit card numbers.
- Mixed format: Email addresses, street addresses, searches.

Text fields can be found in forms, dialog boxes, search, etc.

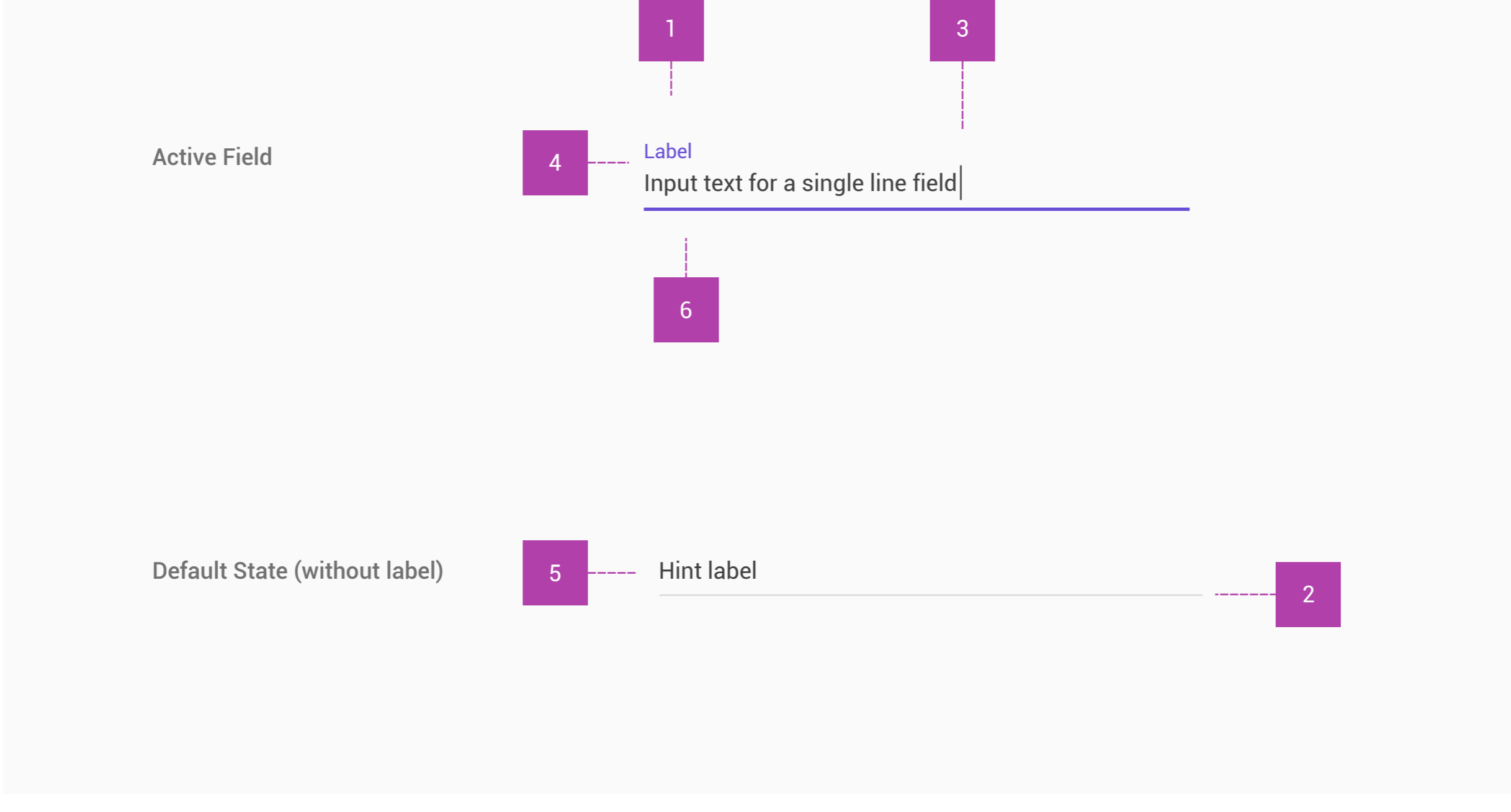
USAGE

Text inputs validate input, help users fix errors, autocomplete words, and provide suggestions.

STRUCTURE

Text fields contain the following elements:

1. Label: Text fields labels display the type of input a field requires.
2. Input line: The input line indicates where to enter text, displayed below the label.
3. Cursor: The cursor indicates the user's current input position.
4. Input text: This refers to the text entered into a field.
 - Input text can be used with autocomplete to help users who have limited literacy or who write in a foreign language.
5. Placeholder text (hint text) : Placeholder text rests in the input field until the user starts entering text. It may contain an action or an example, such as a phone number or email address.
6. Helper text: Helper text gives context about a field's input, such as how the input will be used. (Material Design)



STATES

Text fields have 2 major states: enabled or disabled. Enabled text fields are empty by default.

Disabled text fields are uneducable. They have a dotted input line and less opacity so that they appear less tappable.

In the enabled state, the following user interactions are available:

- Idle/default
- Hover
- Focused

Hover states give the input line more opacity, helping make text fields more noticeable on desktop.

A text field input can either be:

- empty or filled
- valid or invalid

Idle & empty	Label
	Label Helper text
Hover	Label I
	Label I Helper text
Focused	Label Placeholder
	Label Placeholder Helper text
Focus & Filled	Label Input
	Label Input Helper text
Idle & Filled	Label Input
	Label Input Helper text
Invalid & Filled	Label Input Error text
	Label This field is required. Helper text
Disabled & Filled	Label Input
	Label Input
Disabled & Empty	Label
	Label

SPECIFICATIONS

Helper text (optional)

Type: Roboto 12sp

Color: C50

Spacing: 8dp top padding, inset with bottom line

Error Text

Type: Roboto 12sp

Color: R100

Spacing: 8dp top padding, inset with bottom line



State	Bottom Line	Label Text	Input/Placeholder Text
Idle & Empty	Thickness: 1px Color: C50	Type: Roboto 16px Color: C75 Spacing: 16px top padding, 8px bottom padding, inset with bottom line	
Hover	Thickness: 2px Color: C100	Type: Roboto 16px Color: C75 Spacing: 8px bottom padding, inset with bottom line	Type: Roboto 16px Color: C75 Spacing: 8px bottom padding, inset with bottom line
Focused	Thickness: 2px Color: 100% Primary color (V100)	Type: Roboto 12px Color: 75% Primary color (V75) Spacing: 16px top padding and 8px bottom padding	Type: Roboto 16px Color: C75 Spacing: 8px bottom padding, inset with bottom line
Focused & Filled	Thickness: 2px Color: 100% Primary color (V100)	Type: Roboto 12px Color: 75% Primary color (V75) Spacing: 16px top padding and 8px bottom padding	Type: Roboto 16px Color: C100 Spacing: 8px bottom padding, inset with bottom line
Idle & Filled	Thickness: 1px Color: C50	Type: Roboto 12px Color: C75 Spacing: 16px top padding and 8px bottom padding	Type: Roboto 16px Color: C100 Spacing: 8px bottom padding, inset with bottom line
Filled & Invalid	Thickness: 2px Color: R100 (Crimson)	Type: Roboto 12px Color: R100 Spacing: 16px top padding and 8px bottom padding, inset with bottom line	Type: Roboto 16px Color: C100 Spacing: 16dp top padding, 8dp bottom padding
Disabled	Thickness: 1px Color: C50 Dotted	Type: Roboto 12px Color: C50 Spacing: 8px bottom padding, inset with bottom line	Type: Roboto 16px Color: C50 Spacing: 8dp top padding, inset with bottom line

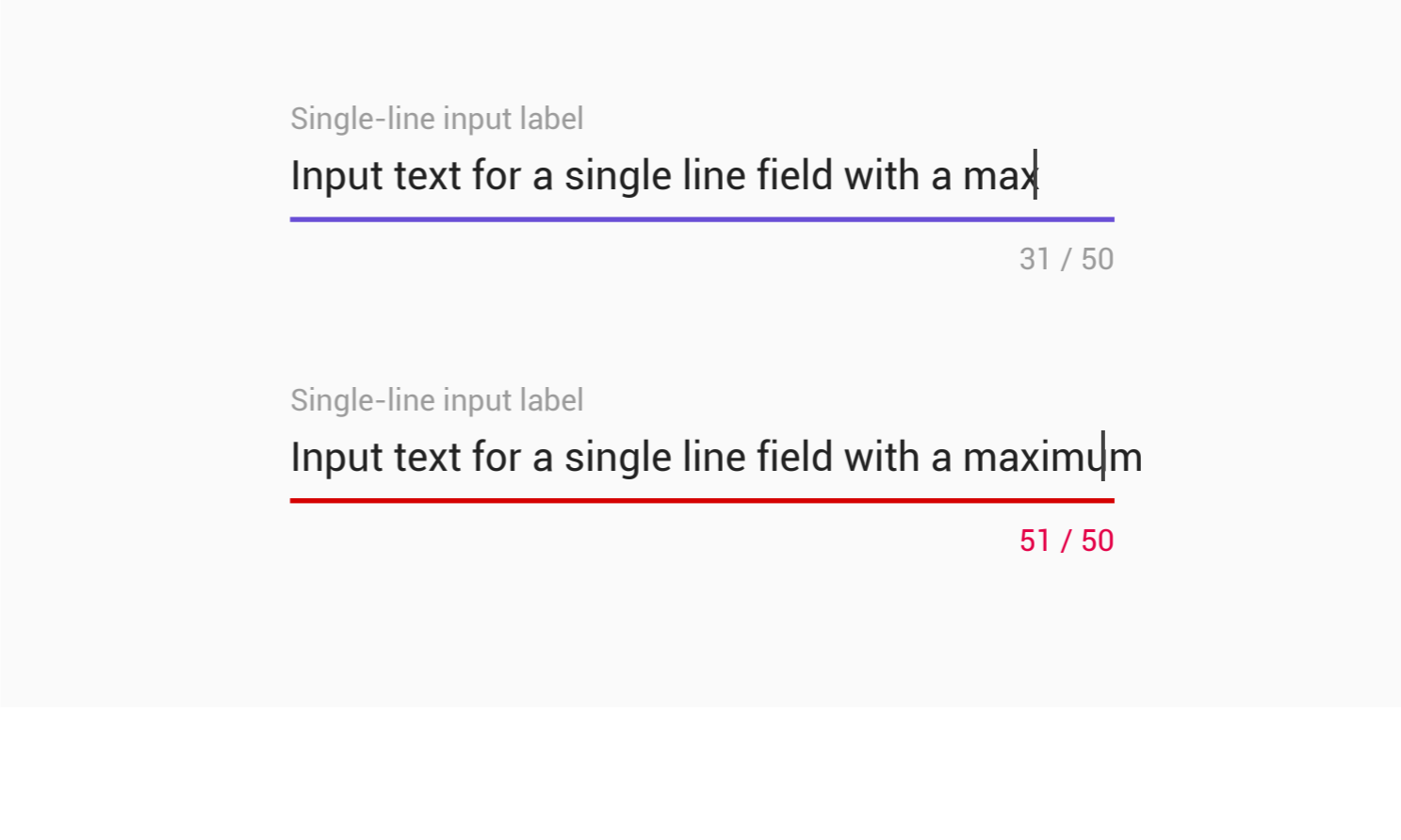
FIELD TYPES

There are several types of text fields:

- Single-line
- Multi-line
- Text area

SINGLE-LINE FIELDS

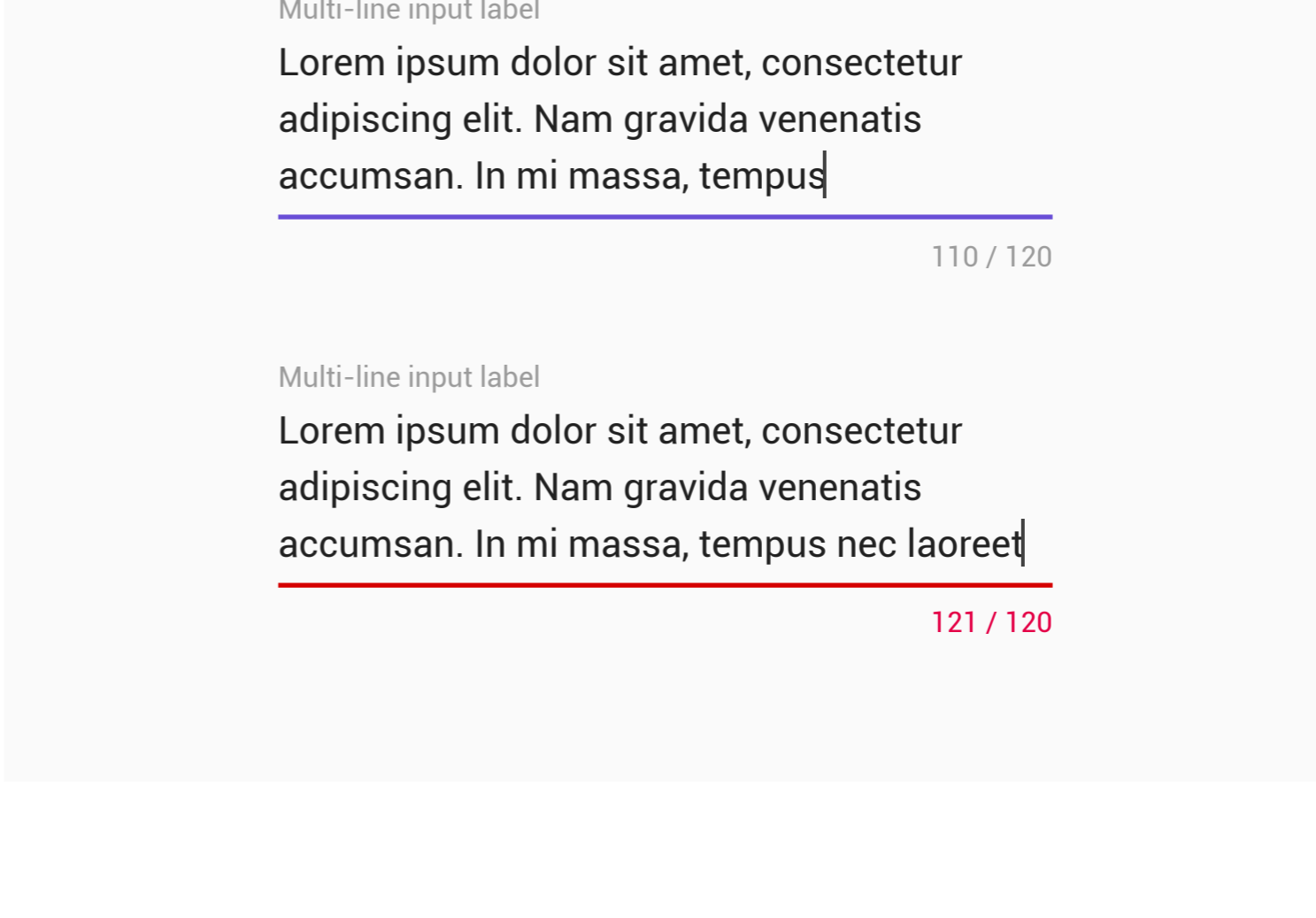
When the text entered is longer than the input line, it automatically scrolls left as the cursor reaches the right edge of the field.



MULTI-LINE FIELDS

Multi-line text fields wrap text onto a new line by expanding the bottom of the field, shifting screen elements downward.

Multi-line text fields allow users to see everything they input at once.



INPUT TYPES

Text fields can be formatted to indicate the types of data a field accepts, using placeholder text, preset layout, and character limits. Text formatting should be displayed only when the field is pressed, focused, or filled.

VARIATIONS

Here are some examples of “other” fields that may have exceptions the guidelines above:

- Search (solo) fields

REASONING

Why have we chosen to implement fields the way we've stated above.

A dropdown arrow indicates that a text field contains nested selections. They display a list of choices, with one choice per line. Each menu item consists of a discrete option or action that can affect the app, the view, or selected elements within a view.

USAGE

Dropdown menus are used mainly for navigation and lists of comments. Dropdown boxes have a dropdown arrow next to them and are used to select attributes or enter form data.

STRUCTURE

A dropdown button selects between multiple selections. The button displays the current state and a down arrow. Available states may be shown as a list of strings, a palette, or icons, for example.

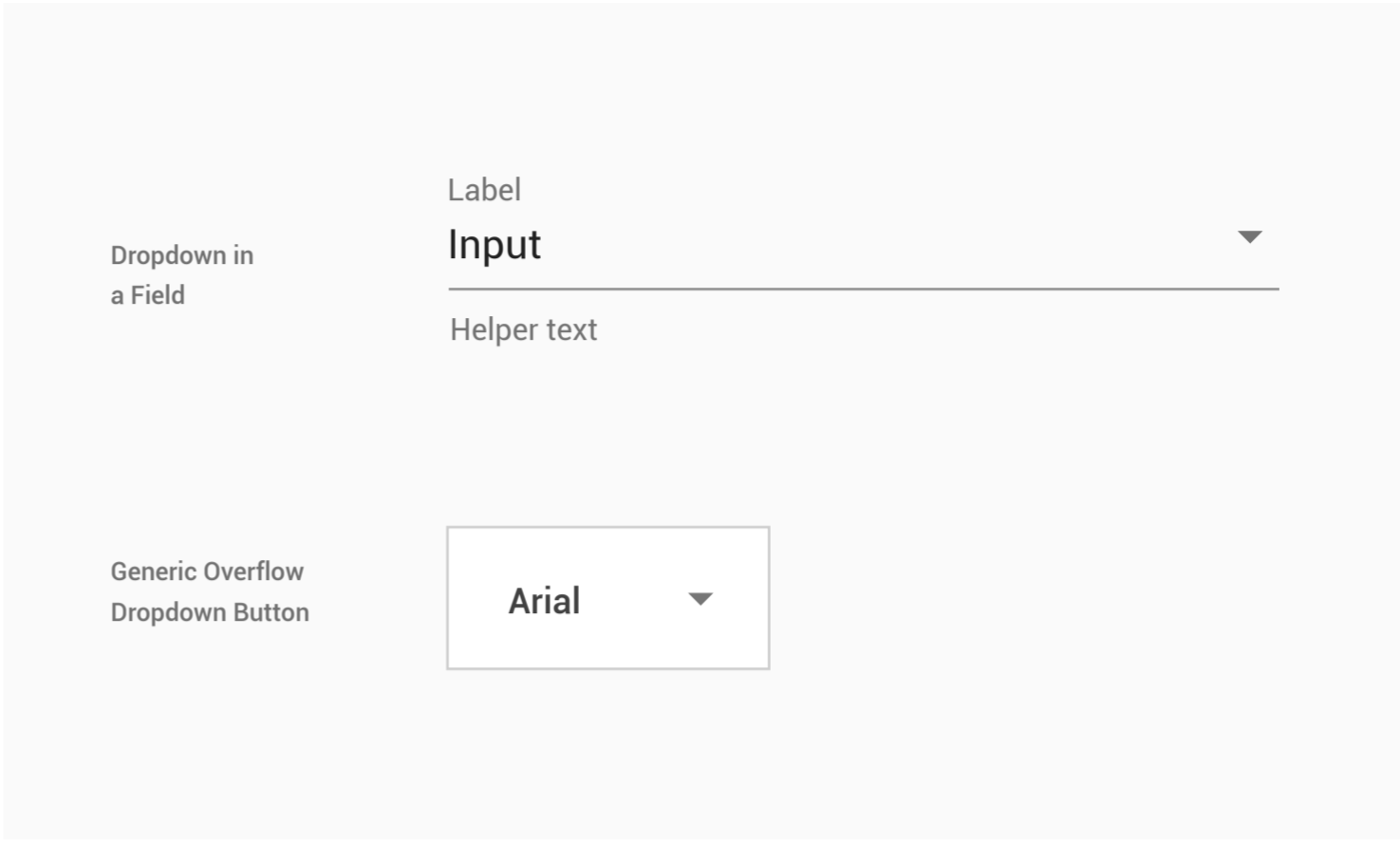
When a user interacts with the button, a menu covers the button and displays the possible states. Pressing a state dismisses the menu and updates the button to display this new state.

Scrolling within the dropdown behaves the same way a menu scrolls.

GENERIC OVERFLOW DROPDOWN BUTTON

The generic overflow dropdown button displays an arrow or menu button by default. When the button is pressed, the menu appears. Pressing an option on the menu navigates to further settings for that option.

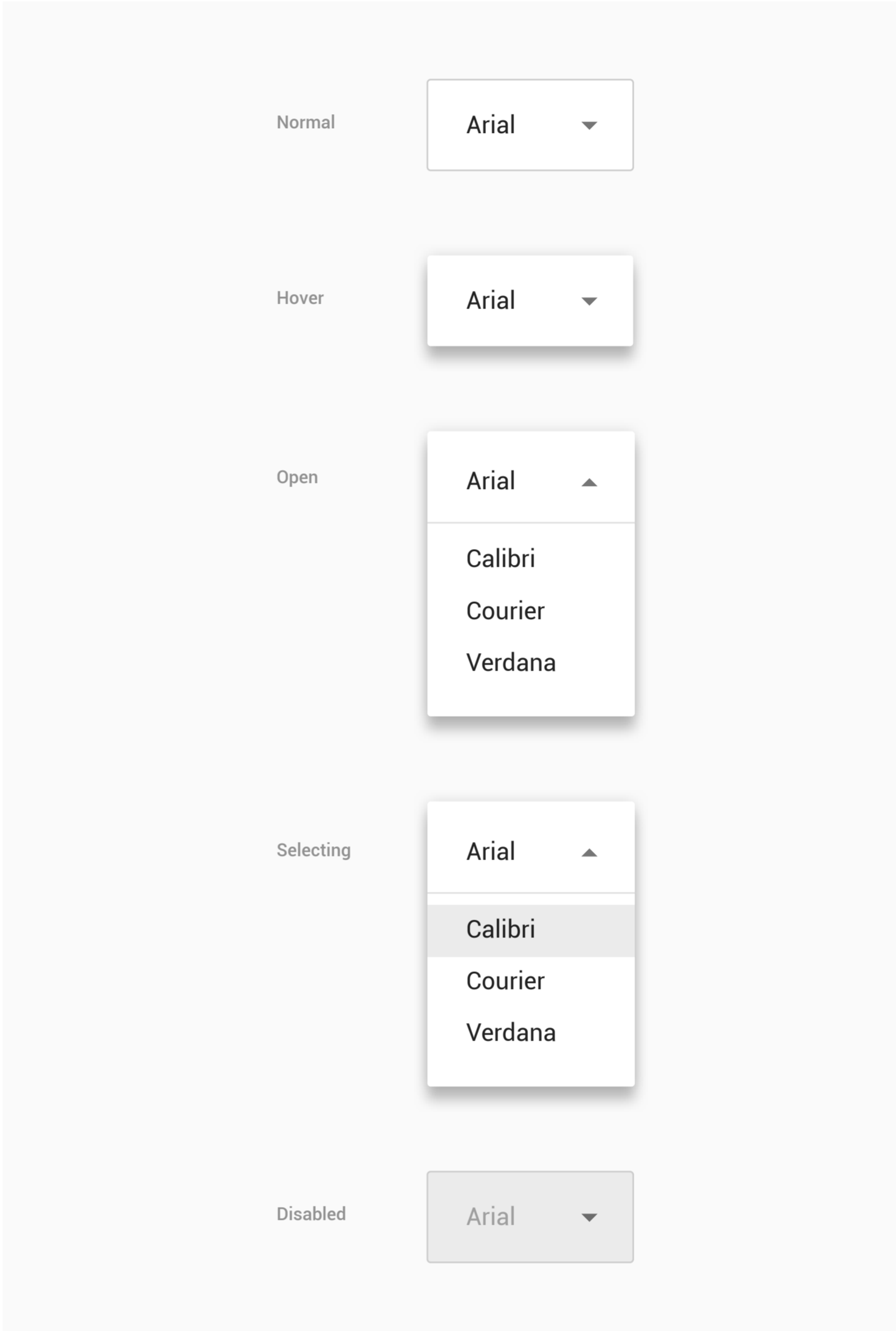
This menu can appear in menus, in fields, or as stand-alone elements.



STATES

Drop-downs have 5 major states:

- Normal
- Hover
- Open
- Selecting
- Disabled



SPECIFICATIONS

Dropdown arrow container

Width: 24px

Height: 24px

Padding below icon: 8px

Icon color: C75

Dropdown button

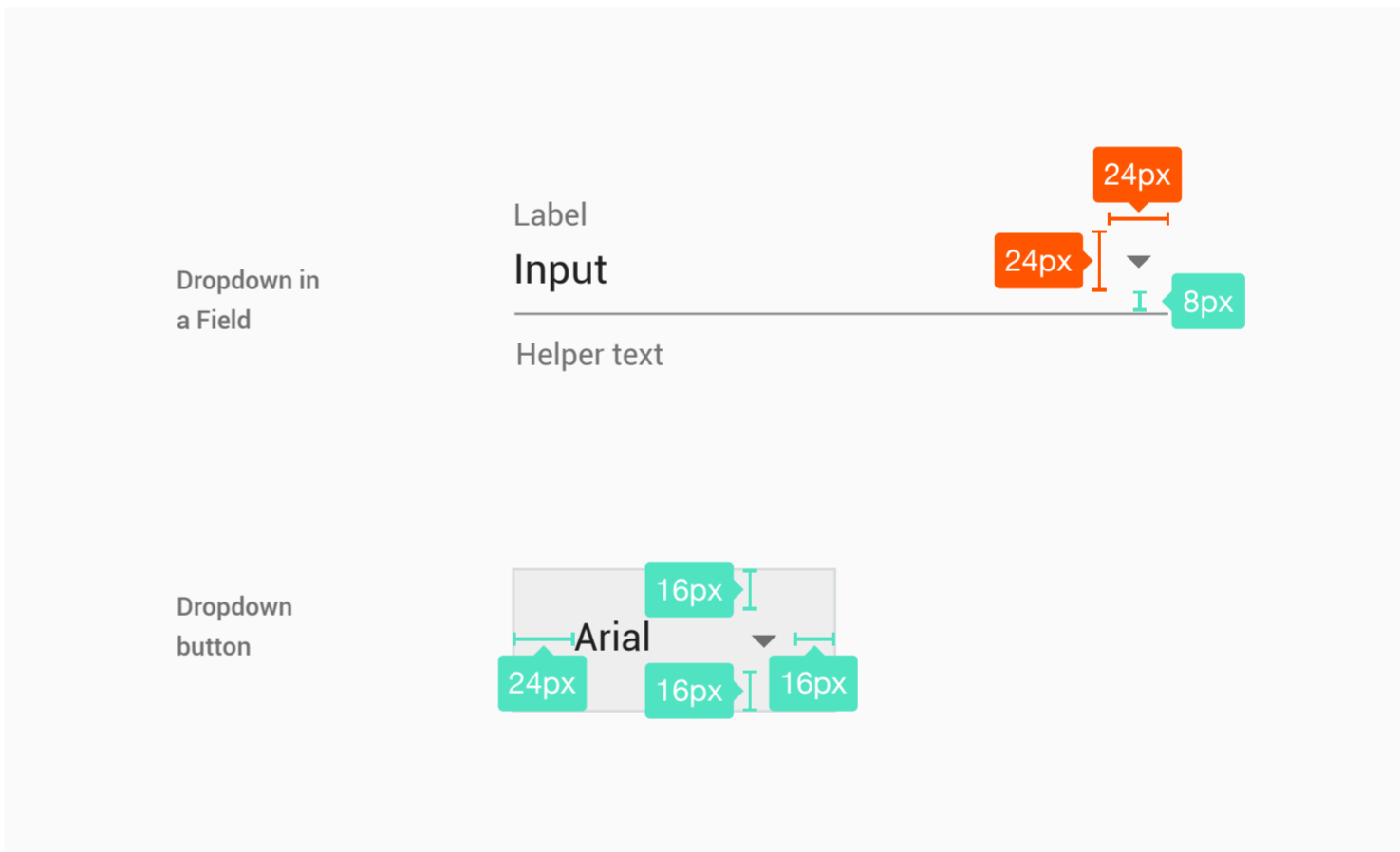
Top and bottom padding: 16px

Left padding: 24px

Right margin: 16px

Horizontal space between font styles: 24px

Corner Radius: 2px



State	Container	Text	Selection
Normal	Border: C25, 1px Color: W100 Elevation: 0px	Type: Roboto 16px Color: C100	
Hover	Border: C25, 1px Color: W100 Elevation: 2px	Type: Roboto 16px Color: C100	
Open	Border: C25, 1px Color: W100 Elevation: 2px	Type: Roboto 16px Color: C100	
Selecting	Border: C25, 1px Color: W100 Elevation: 2px	Type: Roboto 16px Color: C100	Menu item height: 32px Color: C10
Disabled	Border: C25, 1px Color: C10 Elevation: 0px	Type: Roboto 16px Color: C50	

VARIATIONS

Here are some examples of exceptions and variations to the guidelines above.

REASONING

Why have we chosen to implement fields the way we've stated above.

Pickers provide a simple way to select a single value from a pre-determined set.

Date pickers use a dialog window to select a single date on mobile.

Time pickers use a dialog to select a single time (in the hours:minutes format) on mobile. They adjust to a user's preferred time setting.

USAGE

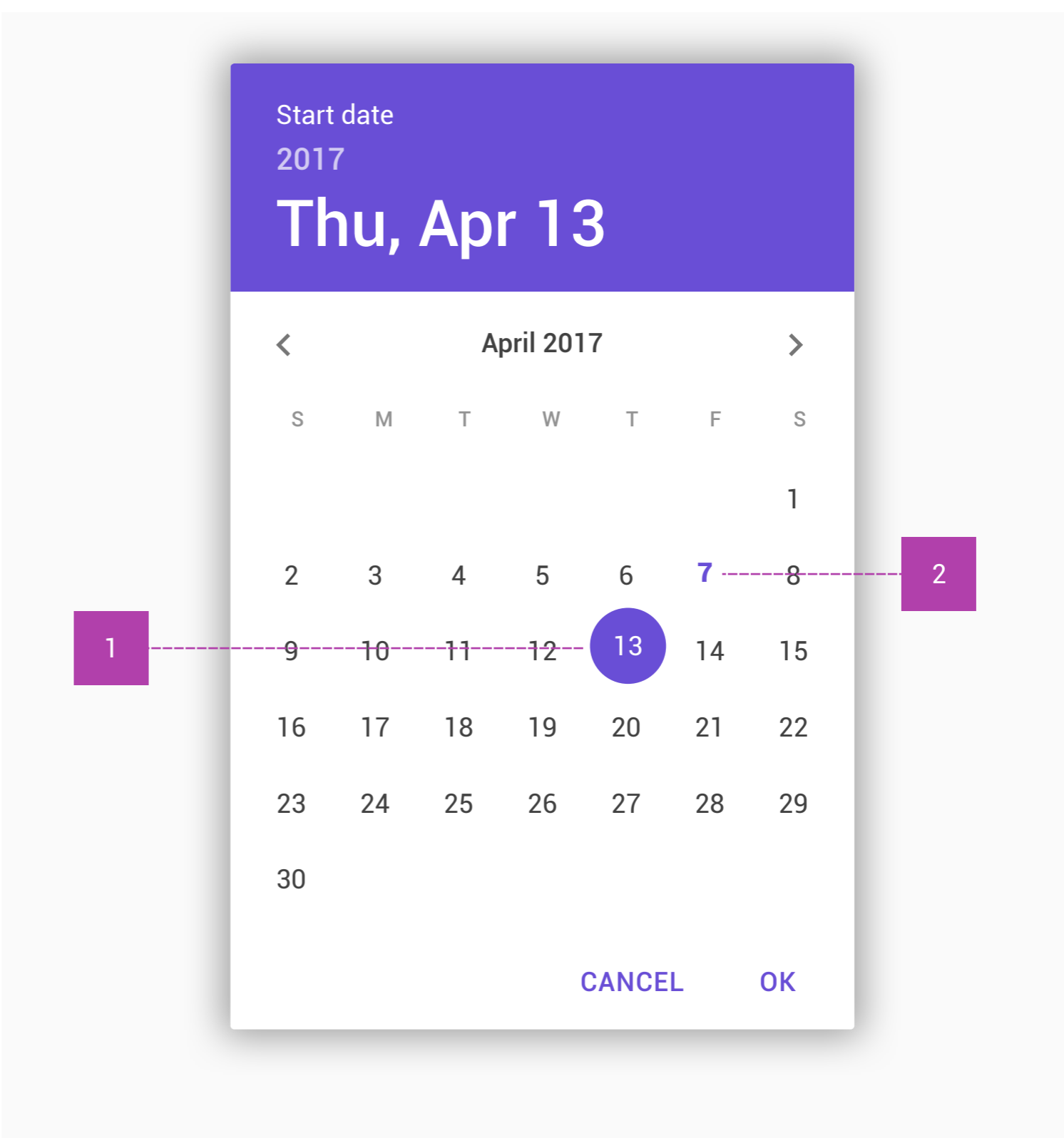
On mobile, pickers are best suited for display in a confirmation dialog.

For inline display, such as on a form, consider using compact controls such as segmented dropdown buttons.

STRUCTURE

DATE PICKER

- These types of pickers are only to be used on mobile platforms.
1. The selected day is indicated by a filled circle.
 2. The current day is indicated by a different color and type weight.
- Users can swipe left to right to navigate through the months. Touching the year in the title bar to transitions to the year view.



SPECIFICATION

DATE PICKER

- 'Start date' text
- Type: Roboto Regular 16px
 - Colour: W100

- Year text:
- Type: Roboto Medium 16px
 - Colour: W100
 - Opacity: 70%

- Current date text:
- Type: Roboto Medium 34px
 - Colour: W100

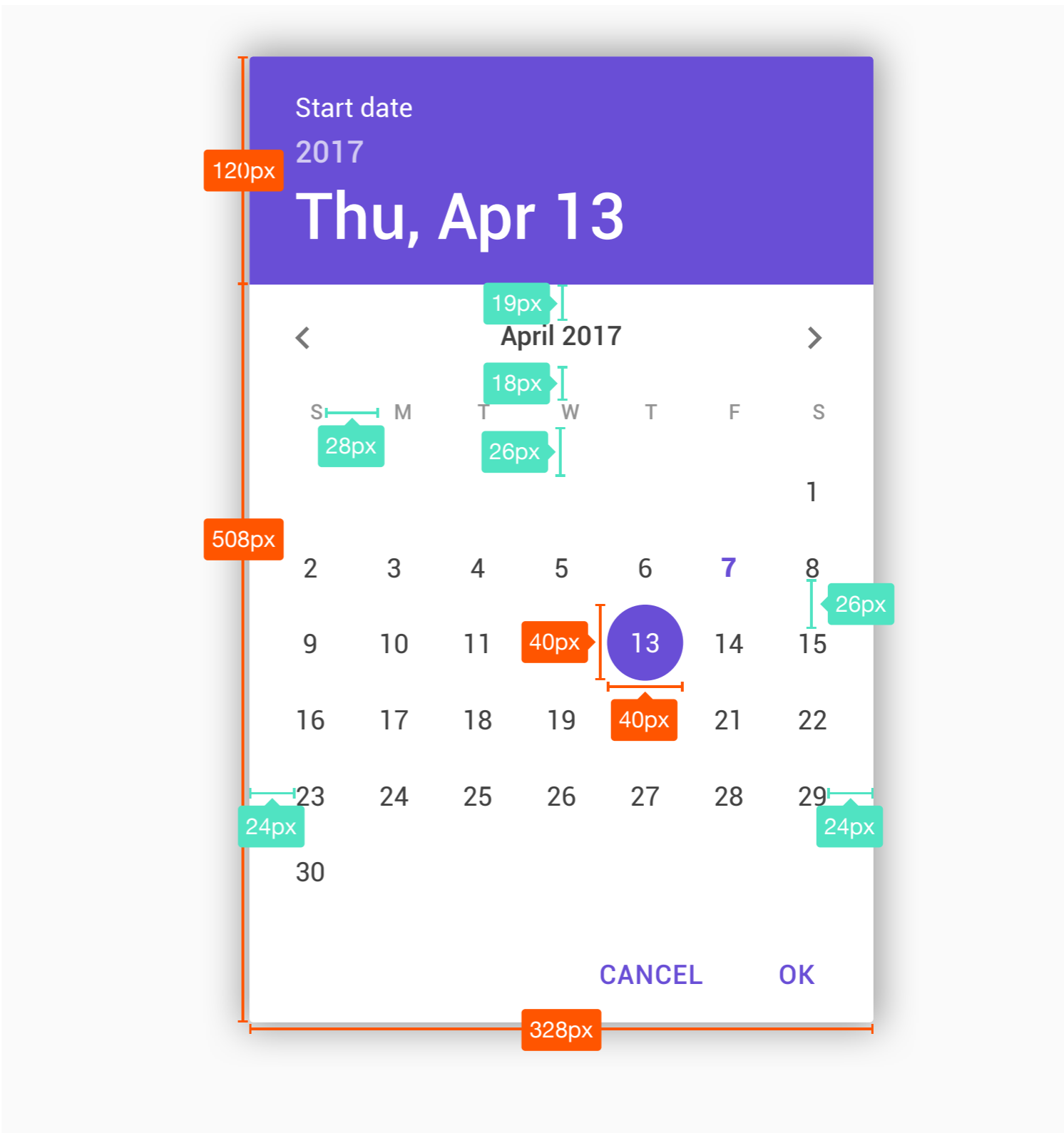
- Current month text:
- Type: Roboto Medium 14px
 - Colour: C100

- Week text:
- Type: Roboto Medium 12pm
 - Colour: C50

- Calendar day text:
- Type: Roboto Regular 12px
 - Colour: C100

- Current day text:
- Type: Roboto Medium 12 px
 - Colour: Primary color (V100)

- Selected day circle:
- Type: Roboto Medium 12 px
 - Colour: Primary color (V100)
 - Top, bottom, left and right text padding: 12px

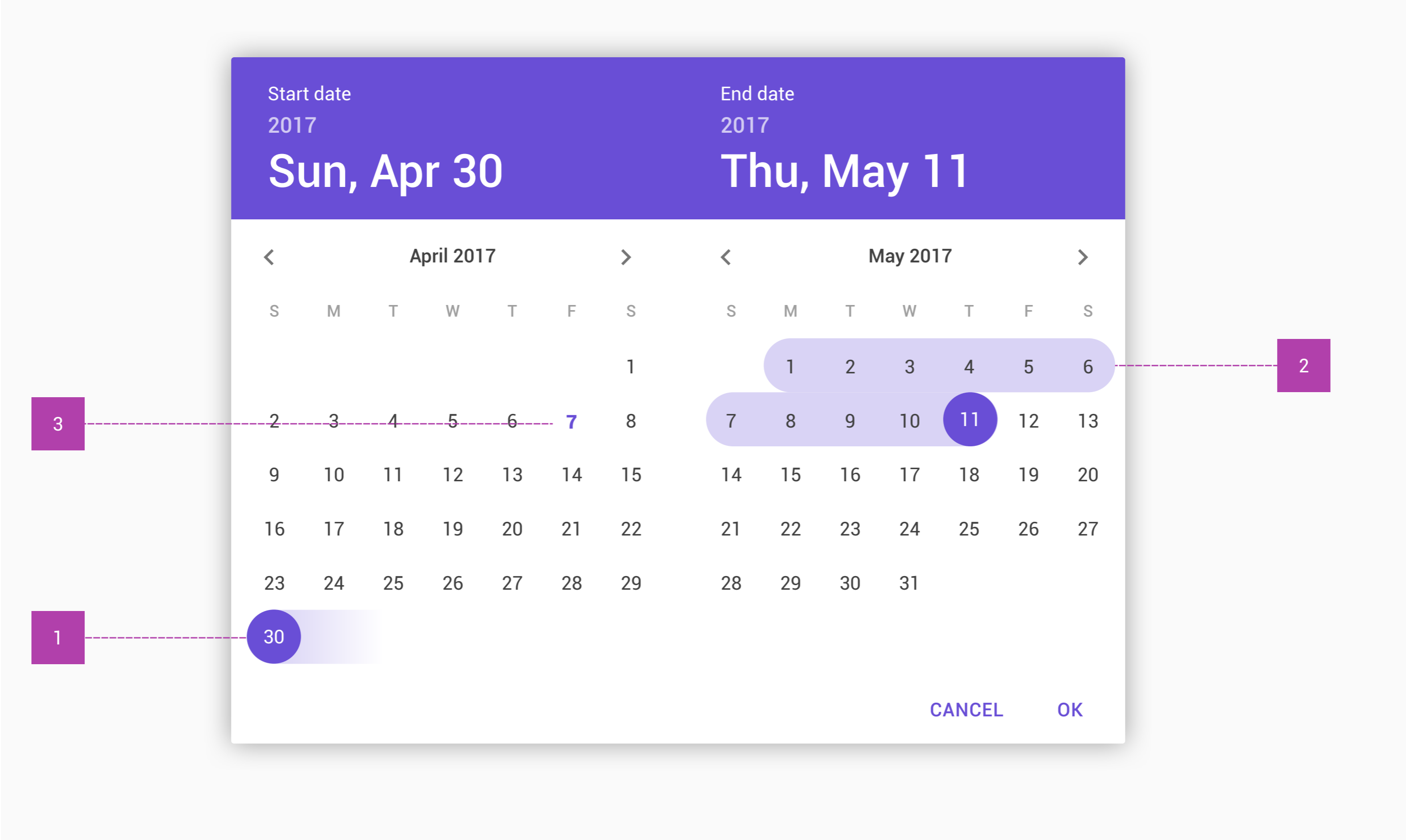


VARIATIONS

Here are some examples of exceptions and variations to the guidelines above.

DATE RANGE PICKER

- These types of pickers are used when a range of dates need to be selected.
1. The selected start and end dates are indicated by a filled circle.
 2. The selected dates in between the start and end dates are indicated by a shaded background.
 3. The current day is indicated by a different color and type weight.
- Users can swipe left to right to navigate through the months. Users begin by selecting the start date and then either dragging to the end date, or clicking on the end date.



REASONING

Why have we chosen to implement pickers the way we've stated above.

Tooltips are text labels that appear when the user hovers over, focuses on, or touches an element.

Tooltips identify an element when they are activated. They may contain brief helper text about its function. For example, they may contain text information about actionable icons.

Tooltip labels do not receive input focus.

Summoned by:

- Hovering over an element with a cursor
- Focusing on an element with a keyboard (usually the tab key)
- Upon touch

USAGE

Use tooltips for interactive imagery.

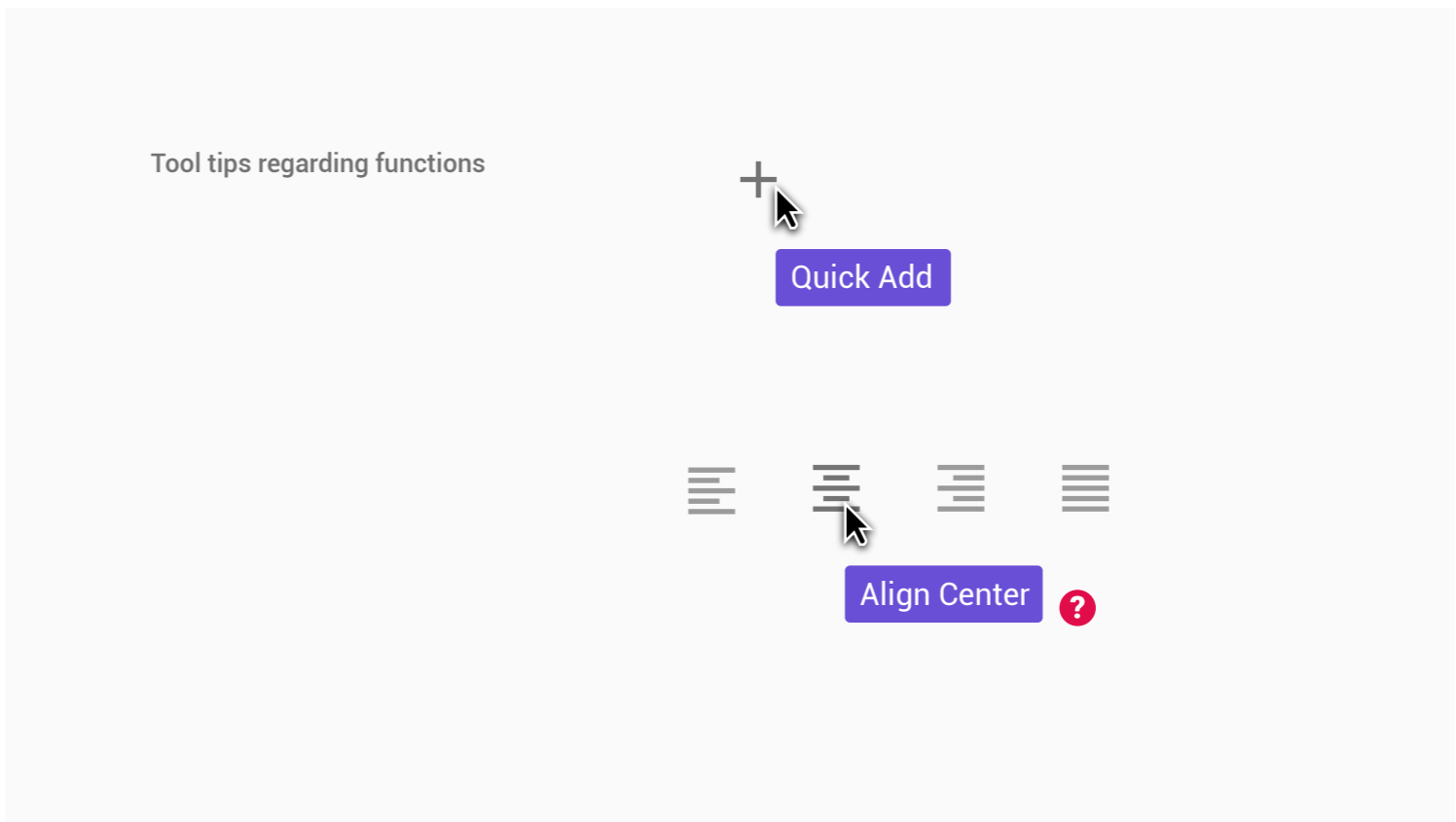
Tooltips don't display rich information including images and formatted text.

Tooltips are different than ALT-attributes, which are intended primarily for static images.

Tooltips don't have directional arrows; instead, they rely on motion emanating from the source to convey direction.

STRUCTURE

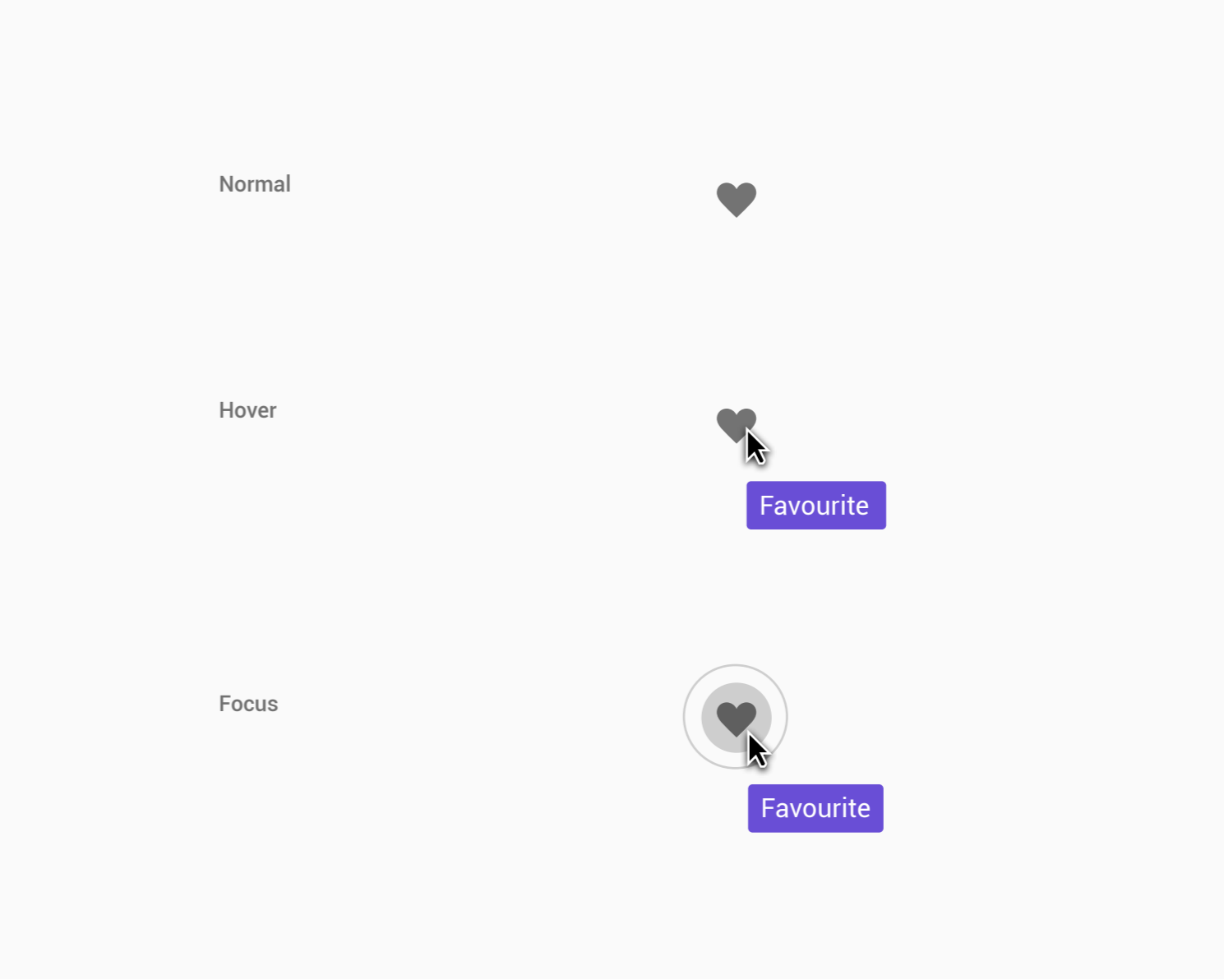
Tool tips appear as white text in a grey container that floats on a layer above all other content when a user interacts with the associated imagery.



6.2 STATES

Tool tip elements have 3 states:

- Normal
- Hover
- Focused/Pressed



SPECIFICATIONS

Text

Text: Roboto Medium 10px

Color: W100

Left and right text padding: 8px

Container

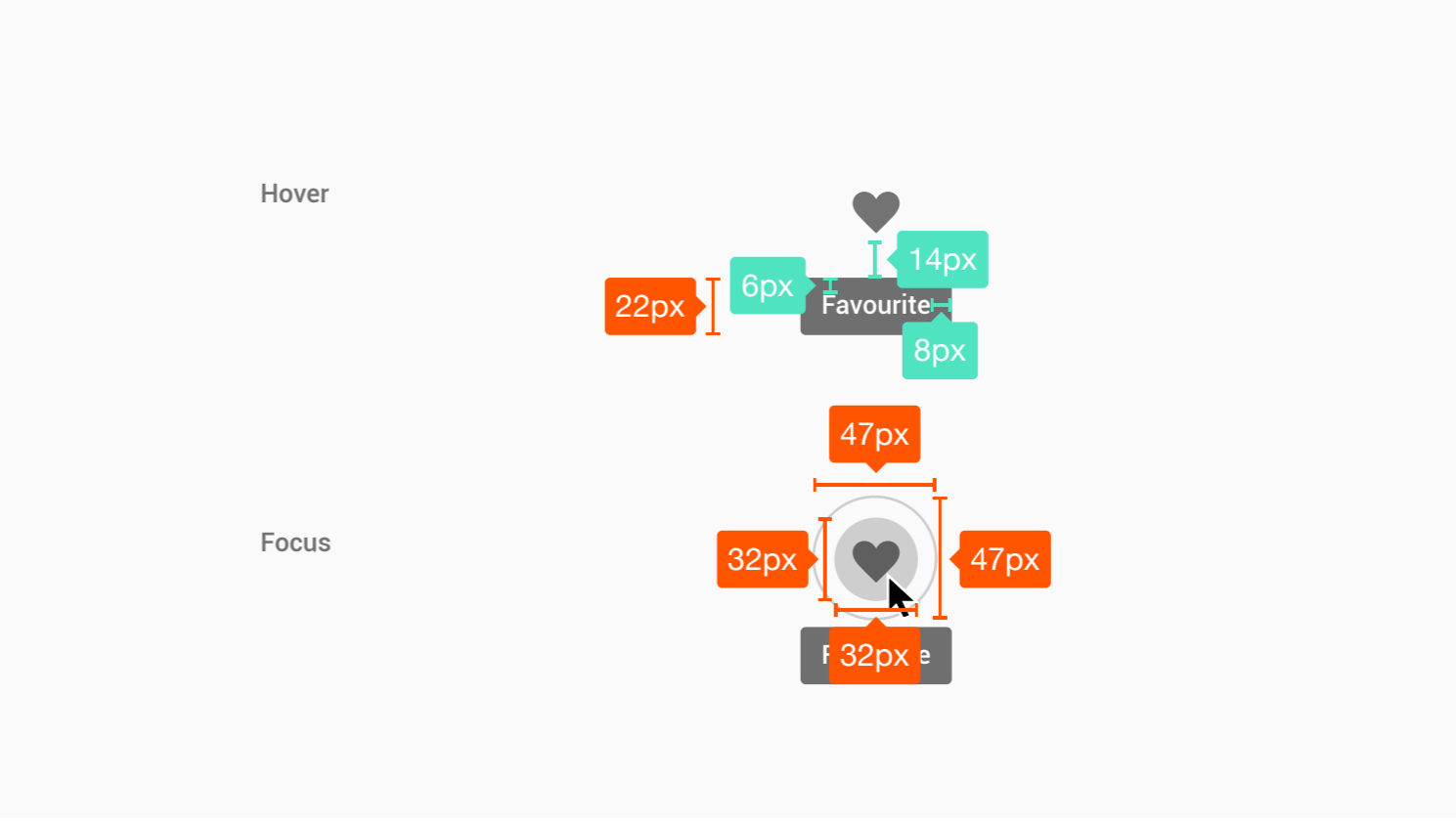
Color: C75

Opacity: 90%

Tile height: 22px

Top margin: 14px

Rounded corners: 2px



INTERACTION

A tooltip is triggered by tapping and holding an item. Keep the tooltip displayed as long as the user continues to hold the element.

Timing

On lift, display the tooltip for 1.5 seconds.

If the user takes another action before that time ends, the tooltip will disappear.

Motion details

The tooltip entrance occurs over 150ms, using the Deceleration Curve. It also exits over 150ms, using the Acceleration Curve.

VARIATIONS

Here are some examples of exceptions and variations to the guidelines above.

REASONING

Why have we chosen to implement fields the way we've stated above.

SELECTIONS

Selection controls allow the user to select options.

Three types of selection controls are covered in this guidance:

- Checkboxes allow the selection of multiple options from a set.
- Radio buttons allow the selection of a single option from a set.
- Switches allow a selection to be turned on or off.

Color

Selection controls use an app's accent color.

Themes

Selection controls are available in both dark and light themes.

CHECKBOXES

USAGE

Checkboxes allow the user to select multiple options from a set.

If you have multiple options appearing in a list, you can preserve space by using checkboxes instead of on/off switches.

If you have a single option, avoid using a checkbox and use an on/off switch instead.

Checked On



Checked Off



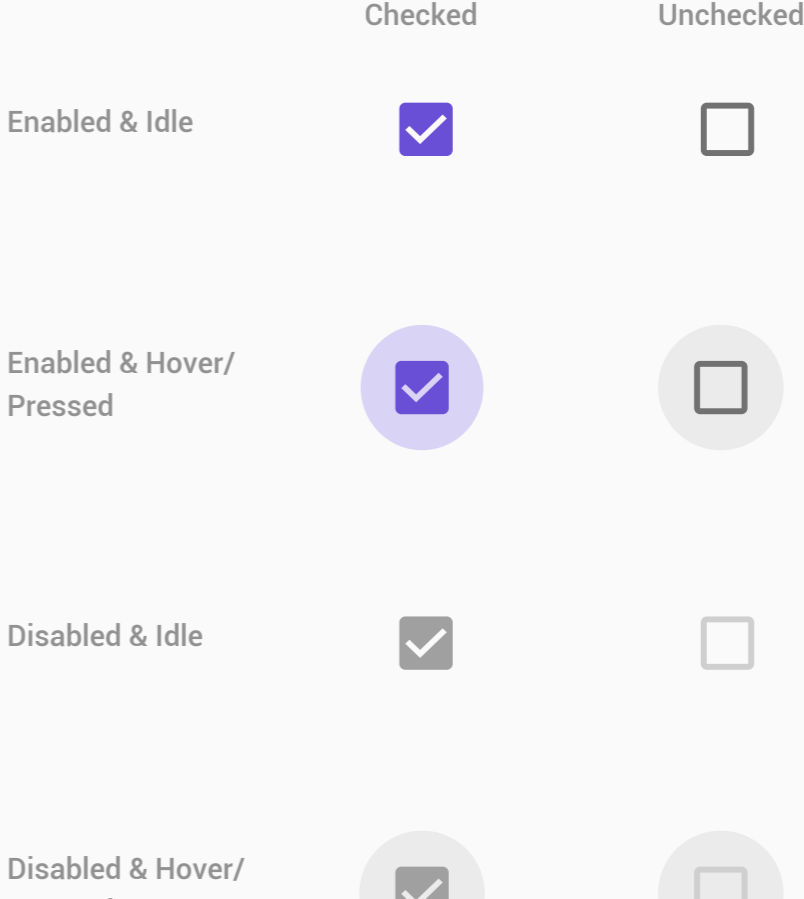
STATES

Checkboxes have 2 major states: enabled or disabled, and can be on or off.

In the enabled and disabled states, the following user interactions are available:

- Idle/default
- Hover/Pressed

Changes from the on to off, or vice versa can only be made from the enabled state.



SPECIFICATIONS

Icon

On: V100

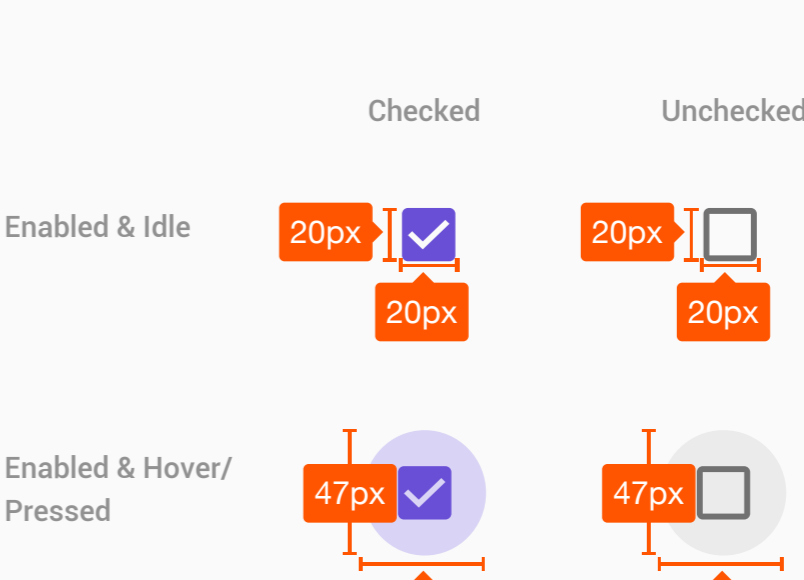
Off: C100

Disabled: C50

Hover/Pressed circle

On: V25

Off: C10



RADIO BUTTONS

USAGE

Radio buttons allow the user to select one option from a set. Use radio buttons for exclusive selection if you think that the user needs to see all available options side-by-side.

Otherwise, consider a dropdown, which uses less space than displaying all options.

On



Off



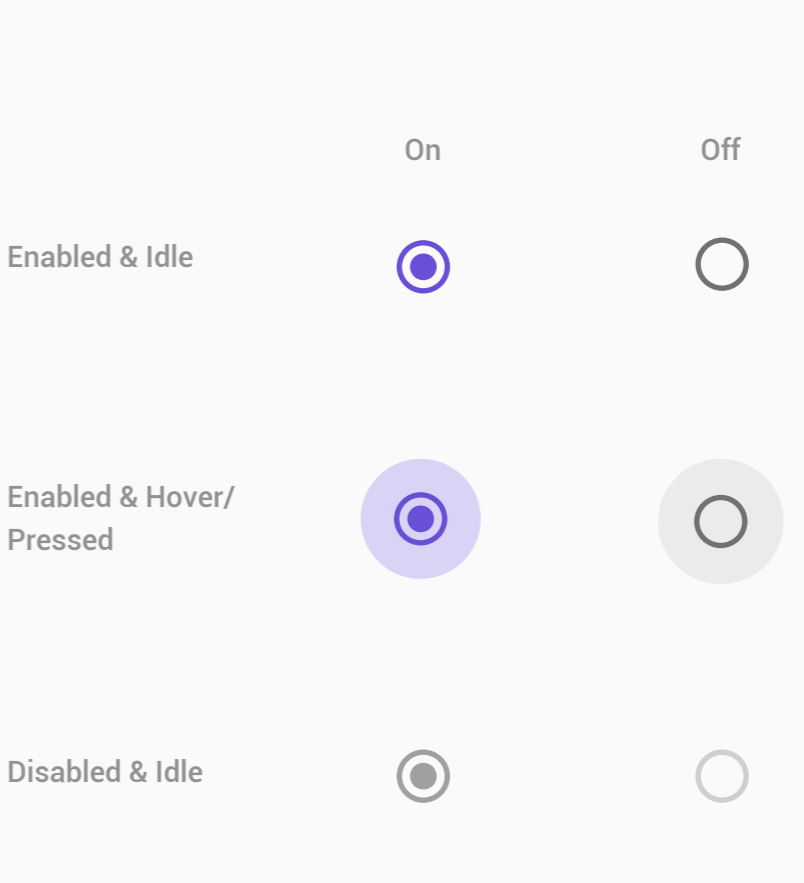
STATES

Radio buttons have 2 major states: enabled or disabled, and can be on or off.

In the enabled and disabled states, the following user interactions are available:

- Idle/default
- Hover/Pressed

Changes from the on to off, or vice versa can only be made from the enabled state.



SPECIFICATIONS

Icon

On: V100

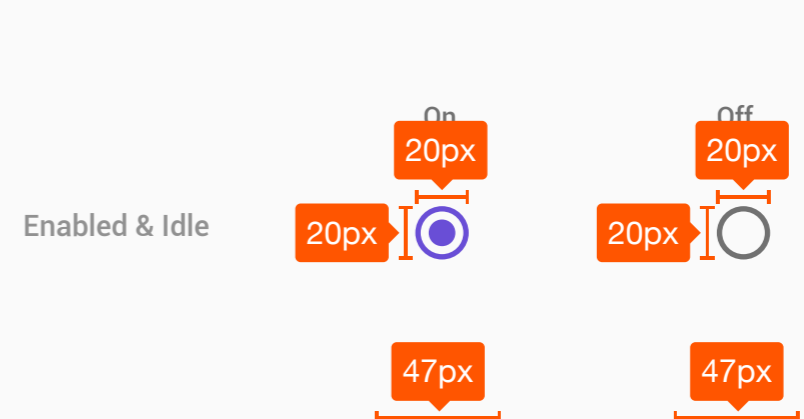
Off: C100

Disabled: C50

Hover/Pressed circle

On: V25

Off: C10



SWITCHES

USAGE

On/off switches toggle the state of a single settings option. The option that the switch controls, as well as the state it's in, should be made clear from the corresponding inline label. Switches take on the same visual properties of the radio button.

The on/off slide toggle with the text "on" and "off" included within the asset is deprecated. Use the switch shown here instead.

On



Off



STRUCTURE

The switch is comprised of 2 pieces:

1. The track
2. The thumb, the circle. The position fo the thumb on the track indicates its state.

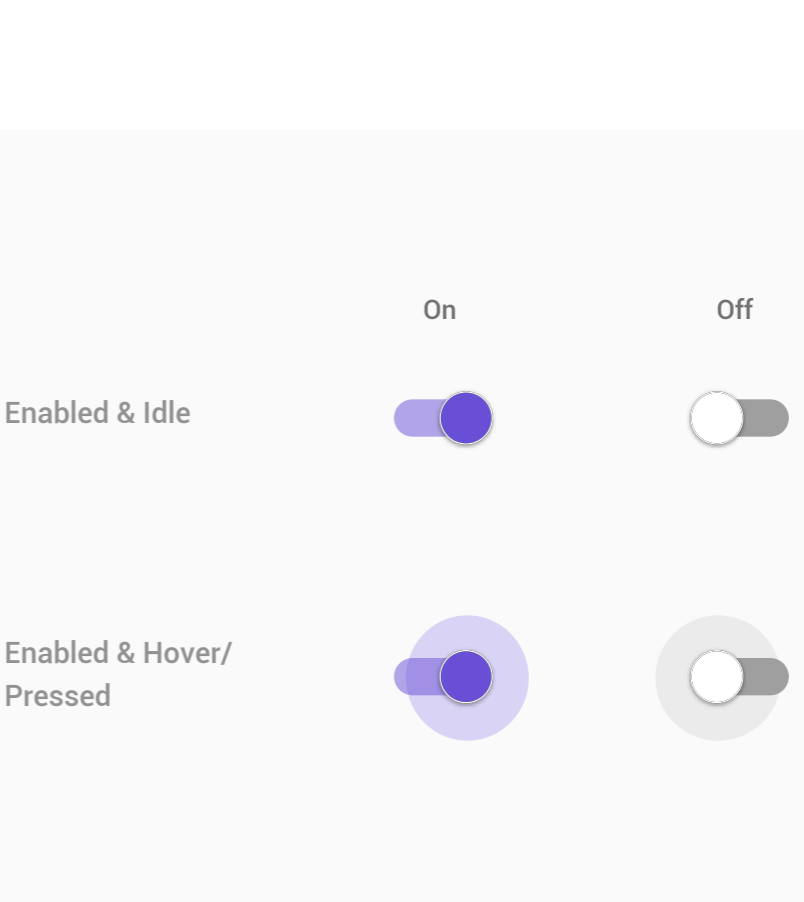
STATES

Radio buttons have 2 major states: enabled or disabled, and can be on or off.

In the enabled and disabled states, the following user interactions are available:

- Idle/default
- Hover/Pressed

Changes from the on to off, or vice versa can only be made from the enabled state.



SPECIFICATIONS

Thumb On: V100

Track On: V50

Hover circle on: V25

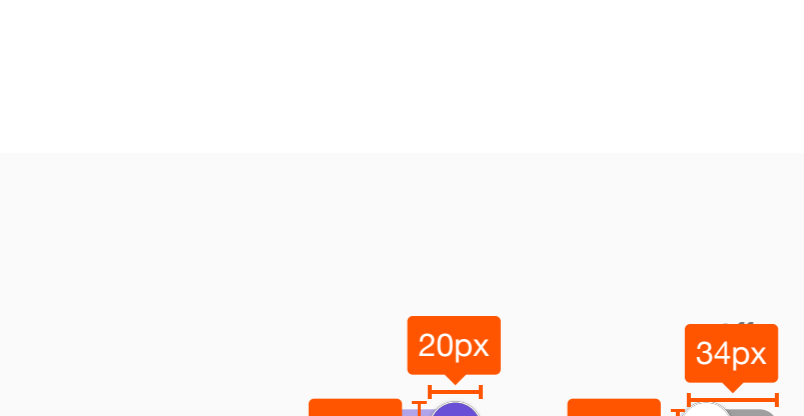
Hover circle off/disabled: C10

Thumb Off: W100

Track Off: C50

Thumb Disabled: C50

Track Disabled: C25



VARIATIONS

Here are some examples of exceptions and variations to the guidelines above.

AS SELECTION CONTROLS

- ☒ Type something
- ☐ Curabitur lobortis id lorem id bibendum. Ut id consectetur magna. Quisque volutpat augue enim, pulvinar
- ☐ Type something
- ☐ Type something
- ☐ Type something

IN DATA TABLES

	Option A	Option B	Option C
Dimension A	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Dimension B	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Dimension C	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>

	Option A	Option B	Option C
Dimension A	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
Dimension B	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>
Dimension C	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>

<input type="checkbox"/> Office	Name	Type	Subtype	Direction
<input type="checkbox"/> DH1	CDB_REQUEST	CDB	IN	Incoming
<input checked="" type="checkbox"/> DH1	LONGNAME_REQUEST	LONGNAME	INOUT	Incoming/Outgoing
<input checked="" type="checkbox"/> DH1	LONGNAME_REQUEST	LONGNAME	INOUT	Incoming/Outgoing
<input type="checkbox"/> DH1	LONGNAME_REQUEST	LONGNAME	INOUT	Incoming/Outgoing
<input type="checkbox"/> DH1	LONGNAME_REQUEST	LONGNAME	INOUT	Incoming/Outgoing

REASONING

Why have we chosen to implement tool tips the way we've stated above.

Buttons communicate the action that will occur when the user touches them.

Material buttons trigger an ink reaction on press. They may display text, imagery, or both. Flat buttons and raised buttons are the most commonly used types.

Primary - Raised button: A typically rectangular material button that lifts and displays ink reactions on press.

Secondary - Bordered button: A button typically rectangular material button but with a white fill background and purple highlight border which makes this element less prominent than the primary option.

Tertiary - Flat button: A button made of ink that displays ink reactions on press but does not lift. Shaped as an invisible rectangle, a label with a transparent bounding box.

USAGE

They may be used inline.

Buttons are used primarily on action items. Some examples include Add, Save, Delete, Sign up. Do not use Buttons as navigational elements. Instead, use Links because it takes the user to a new page and is not associated with an action. Each page may have one to two primary buttons. Any remaining calls-to-action are represented as secondary buttons.

RECOMMENDED BUTTON PLACEMENT

Forms

Button alignment on screen: Left

Place the affirmative button on the left, the dismissive button on the right.

Cards

Buttons are best placed on the left side of a card to increase their visibility. However, as cards have flexible layouts, buttons may be placed in a location suited to the content and context, while maintaining consistency within the product.

Non-standard dialogs and modal windows

Button placement in non-standard dialogs and modal windows depends on the complexity of the content they contain.

For dialogs with relatively simple content, it is recommended to place buttons on the right side of a dialog, with the affirmative button to the right of the dismissive button.

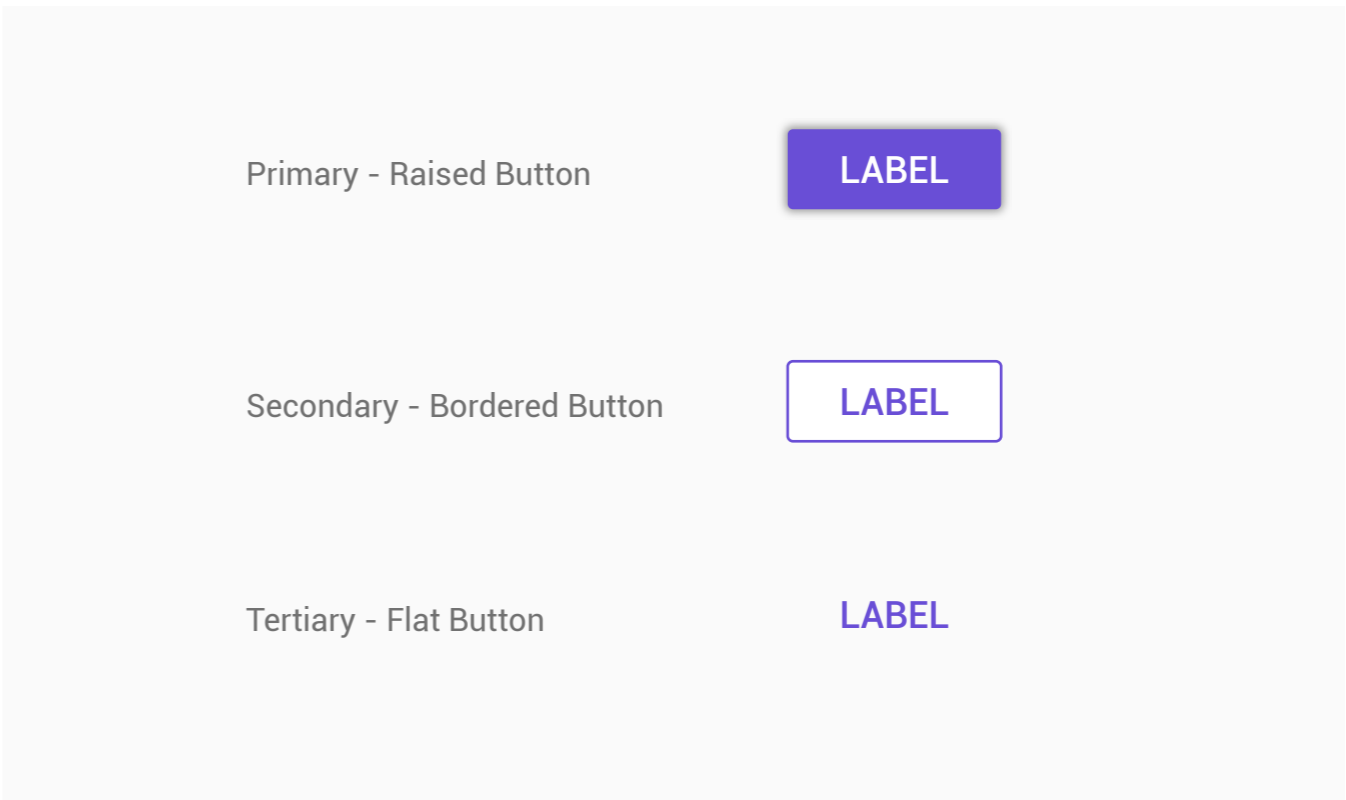
For lengthy or complex forms, it is recommended to place buttons on the left of the form, with the affirmative button to the left of the dismissive button.

STRUCTURE

Buttons are composed of a label and a container.

Button labels tell users what will happen when they click the button. Use verbs that describe the action, such as Add or Delete. Use sentence-style capitalization (only the first word in a phrase and any proper nouns capitalized) and no more than three words for button labels.

For Sets of Buttons, use specific labels, such as Save or Discard, instead of using OK and Cancel. This is particularly helpful when the user is confirming an action.

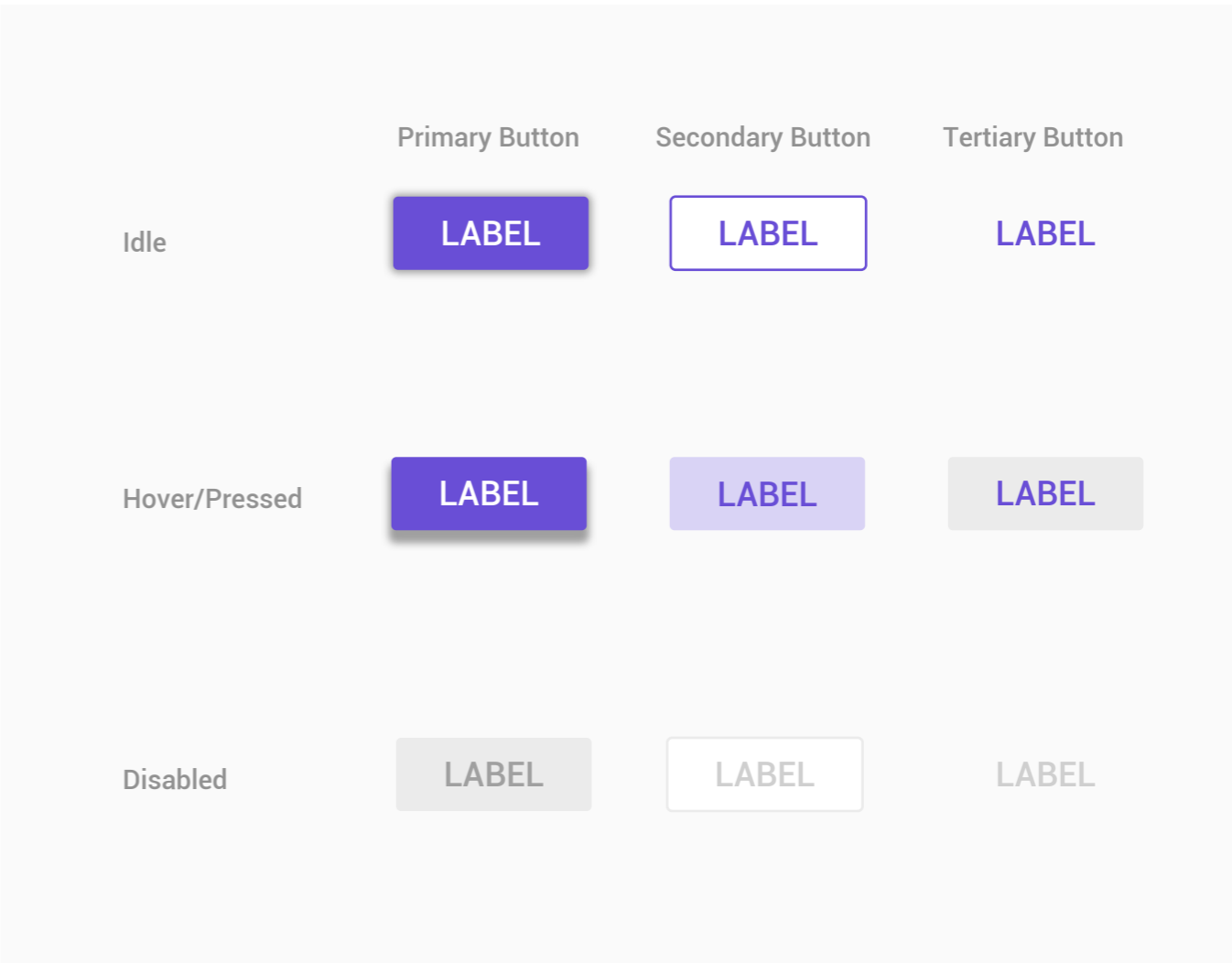


STATES

All buttons have 3 different states:

- Idle/default
- Hover/Pressed
- Disabled

Disabled buttons are used when the user cannot proceed until an input is collected. ?



SPECIFICATIONS

Buttons

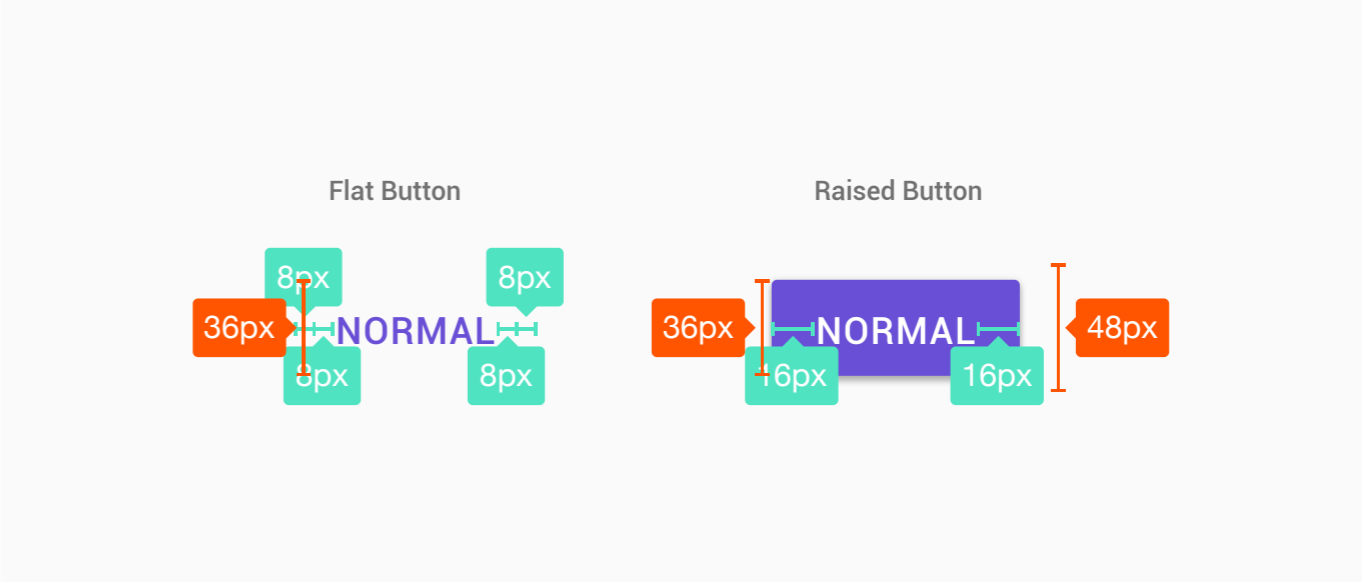
Button text: 14px Roboto medium

Button height: 36px

Minimum width: 88px

Corner radius: 2px

Touch target height: 48dp



Primary Button

Button text left and right padding: 16dp

Touch target height: 48dp

Pressed/hover elevation: 2px

Disabled text: C25

Disabled button: C10

Secondary Button

Horizontal margin: 8dp

Horizontal padding: 8dp

Disabled text: C25

VARIATIONS

Here are some examples of exceptions and variations to the guidelines above.

REASONING

Why have we chosen to implement tool tips the way we've stated above.

A hyperlink, or simply a link, is a reference to data that the reader can directly follow either by clicking, tapping, or hovering. A hyperlink points to a whole document or to a specific element within a document. Hypertext is text with hyperlinks. The text that is linked from is called anchor text.

USAGE

Use text for links rather than graphics or icons.

Links should be three words or less.

Text should be consistent with the title of the intended destination.

Use caution with links that are several words long. It is recommended that links are long enough to be understood by the user, but short enough to prevent text wrapping.

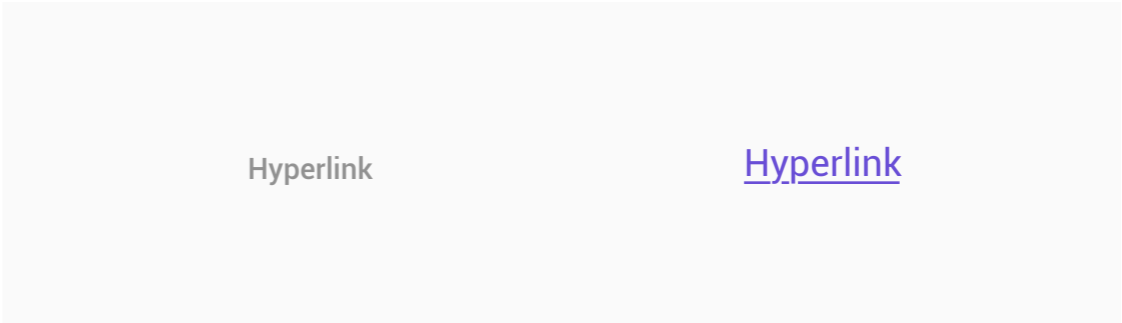
Avoid the term “Click here,” other links to “here,” or the web address itself.

Instead, use a meaningful descriptive label for the link, and match the destination site name.

STRUCTURE

Link text should be set in set in sentence case with the first letter of each word capitalized. Links should not exceed three words.

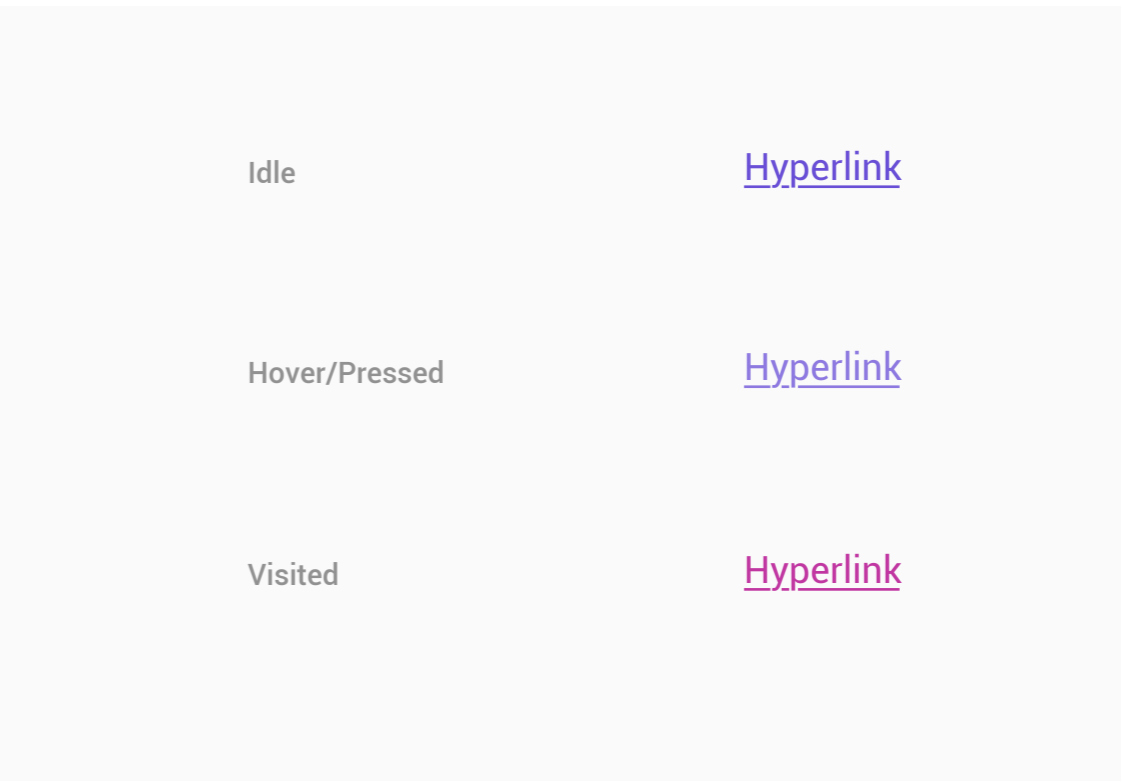
Links can be grouped horizontally or vertically, and must be underlined.



STATES

Hyperlinks have 3 main states

- Idle/default
- Hover/Pressed
- Visited



SPECIFICATIONS

Text

Idle text: 14px Roboto

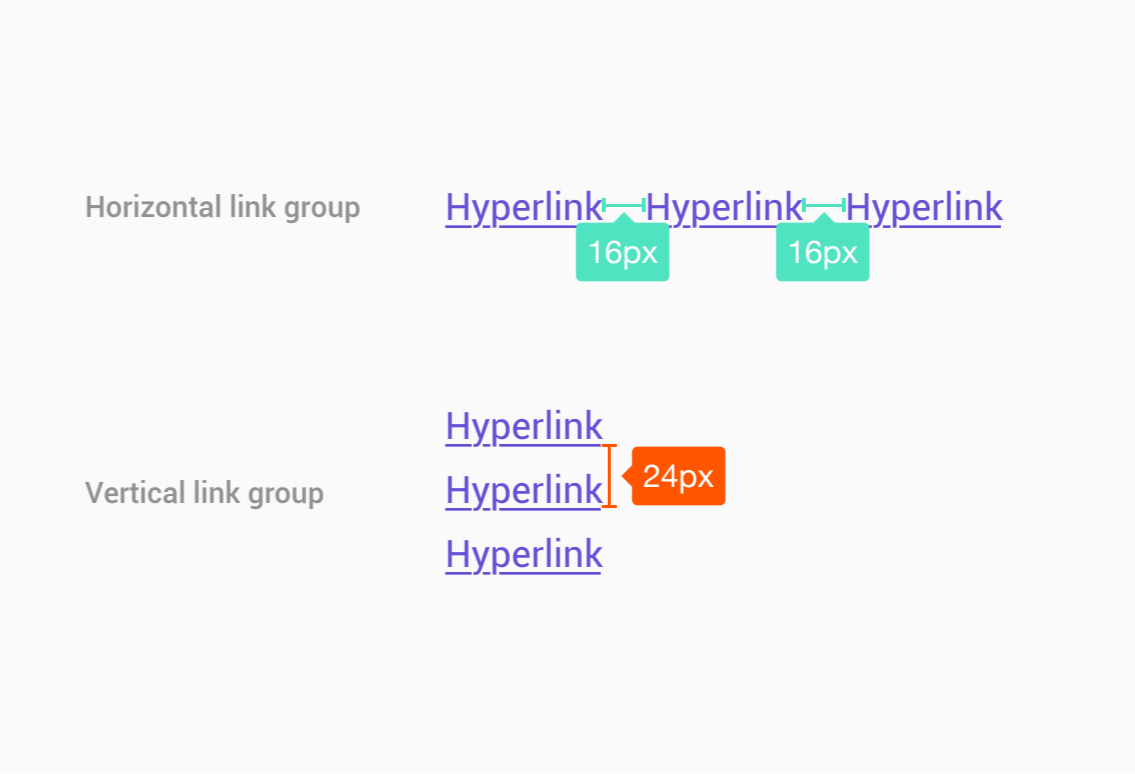
Idle text color: V100

Hover text: 14px Roboto

Hover text color: V75

Idle text: 14px Roboto

Idle text color: F100



Link Groups

Line height: 24 px

Horizontal spacing: 16px

VARIATIONS

Here are some examples of exceptions and variations to the guidelines above.

REASONING

Why have we chosen to implement tool tips the way we've stated above.

Toggle buttons may be used to group related options. Arrange layout and spacing to convey that certain toggle buttons are part of a group.

Focus and pressed states may reinforce that toggles are part of a group. For example, when one button segment is focused, focus may be displayed simultaneously on the rest of the segment.

Toggle button requirements:

- Have at least three toggle buttons in a group
- Label buttons with text, an icon, or both

The following combinations are recommended:

- Multiple and unselected
- Exclusive and unselected
- Exclusive only

USAGE

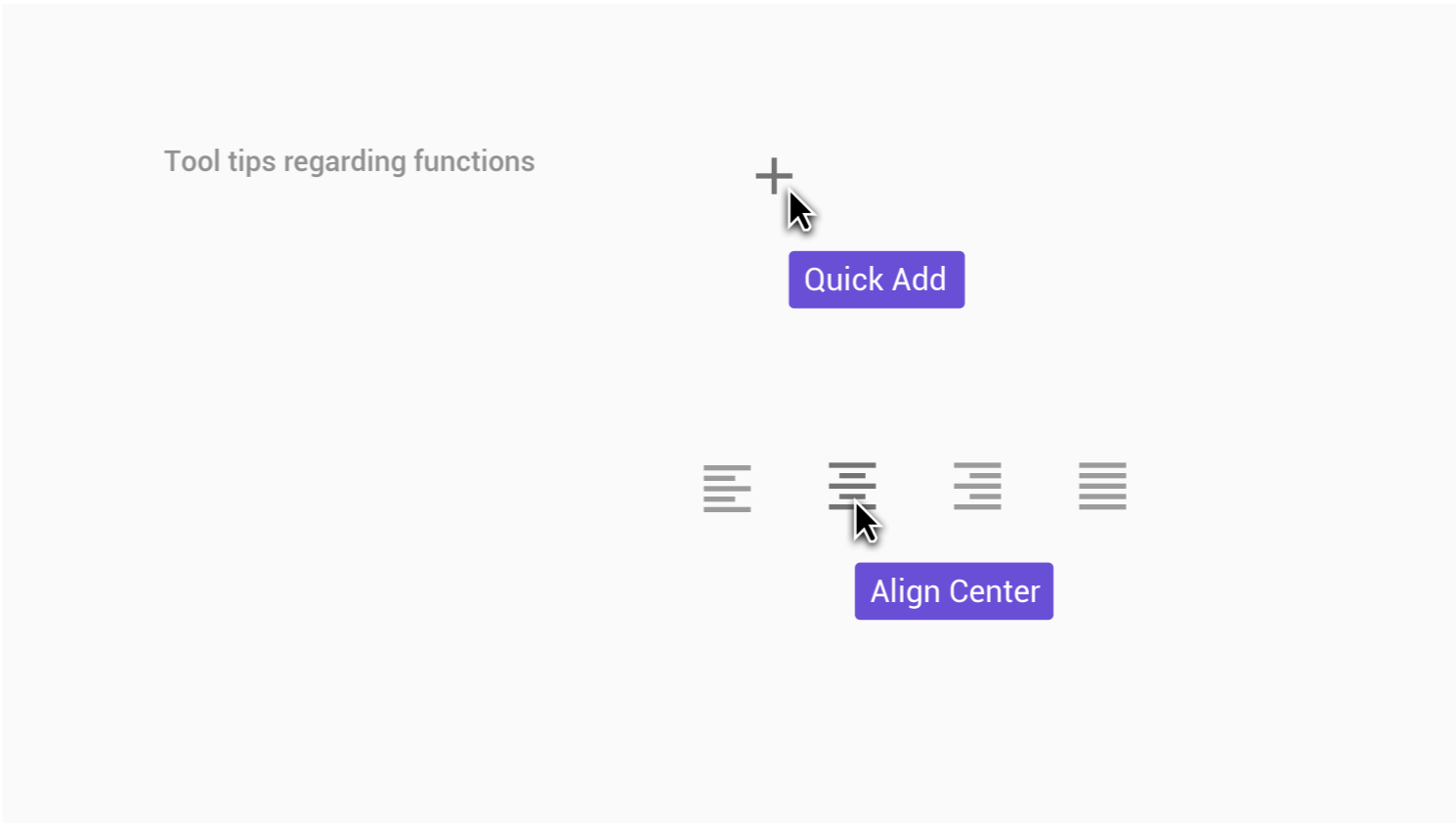
Icons are appropriate for toggle buttons that allow a single choice to be selected or deselected, such as adding or removing a star to an item.

Icon toggles may display bounded or unbounded ink surface reaction ripples beyond their touch-target bounds.

They are best located in app bars, toolbars, action buttons or toggles.

STRUCTURE

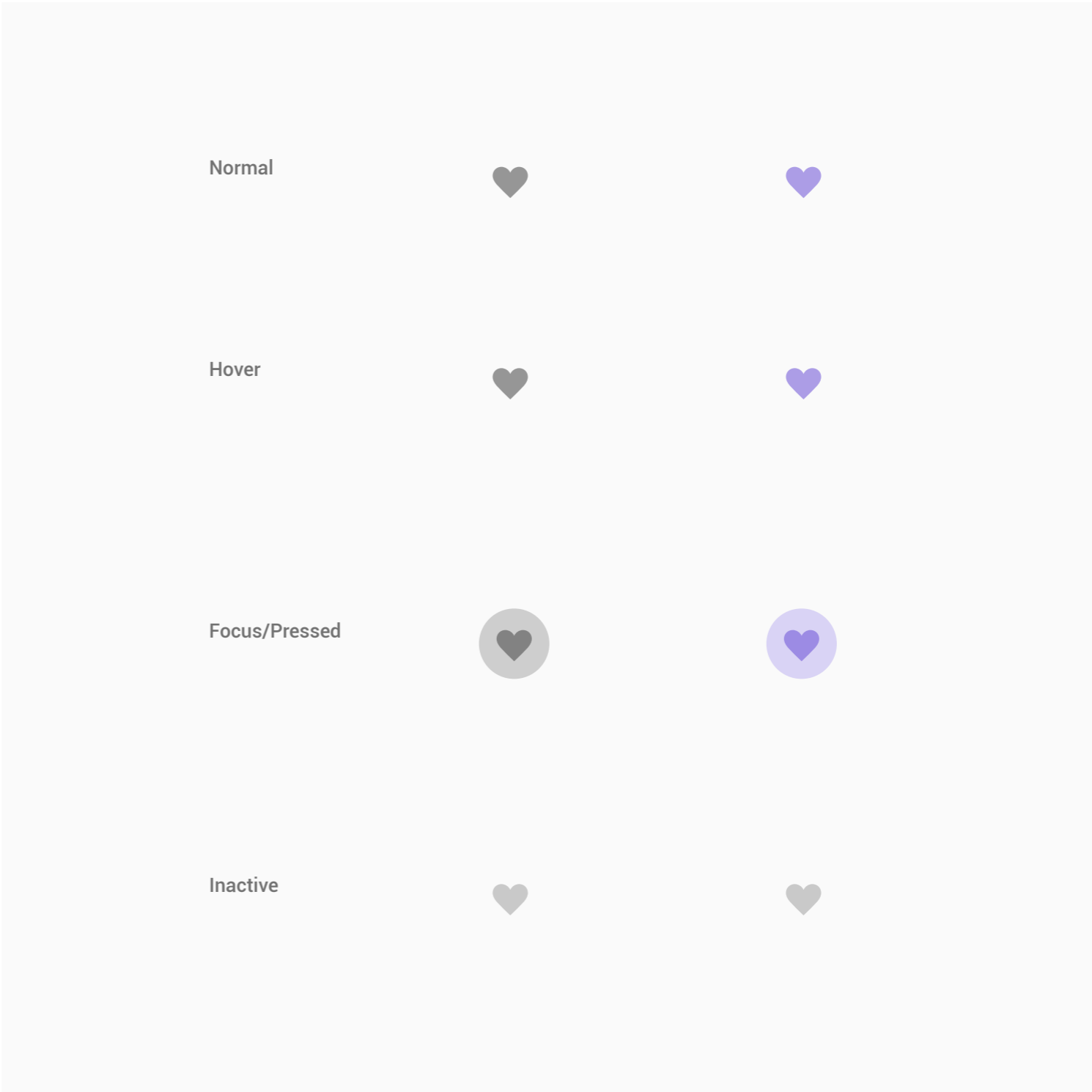
Icon buttons are comprised of an icon and a label, either a persistant label or a hidden label as a tooltip.



STATES

Tool tip elements have 4 states:

- Normal
- Hover
- Focused/Pressed
- Inactive



SPECIFICATIONS

Icon size: 24px x 24px

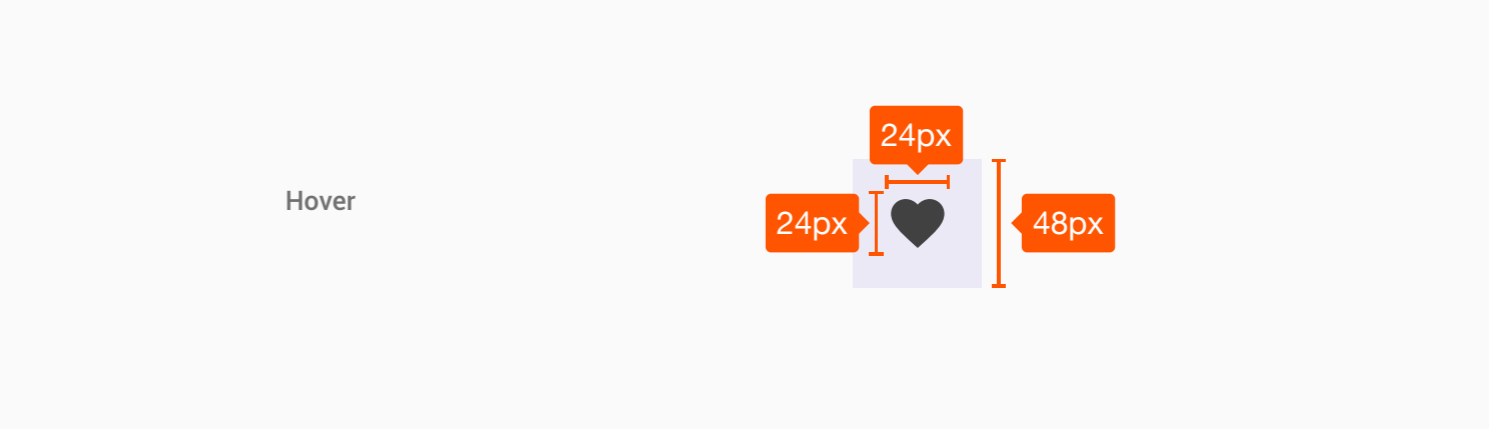
Icon shadow size: 36px x 36px

Touch target: 48px x 48px

Inactive icon color: C100

Active icon color: 100% active color

Disabled color: C50



VARIATIONS

Here are some examples of exceptions and variations to the guidelines above.

REASONING

Why have we chosen to implement tool tips the way we've stated above.

TAGS & COUNTERS



Our visual signature for interfaces is based on cards which create the containers for any information displayed on the screen.

APPROVED

INCOMING

VERSION 1

VERSION 1

5

2

Steppers display progress through a sequence of logical and numbered steps. They may also be used for navigation.

Steppers may display a transient feedback message after a step is saved.

USAGE

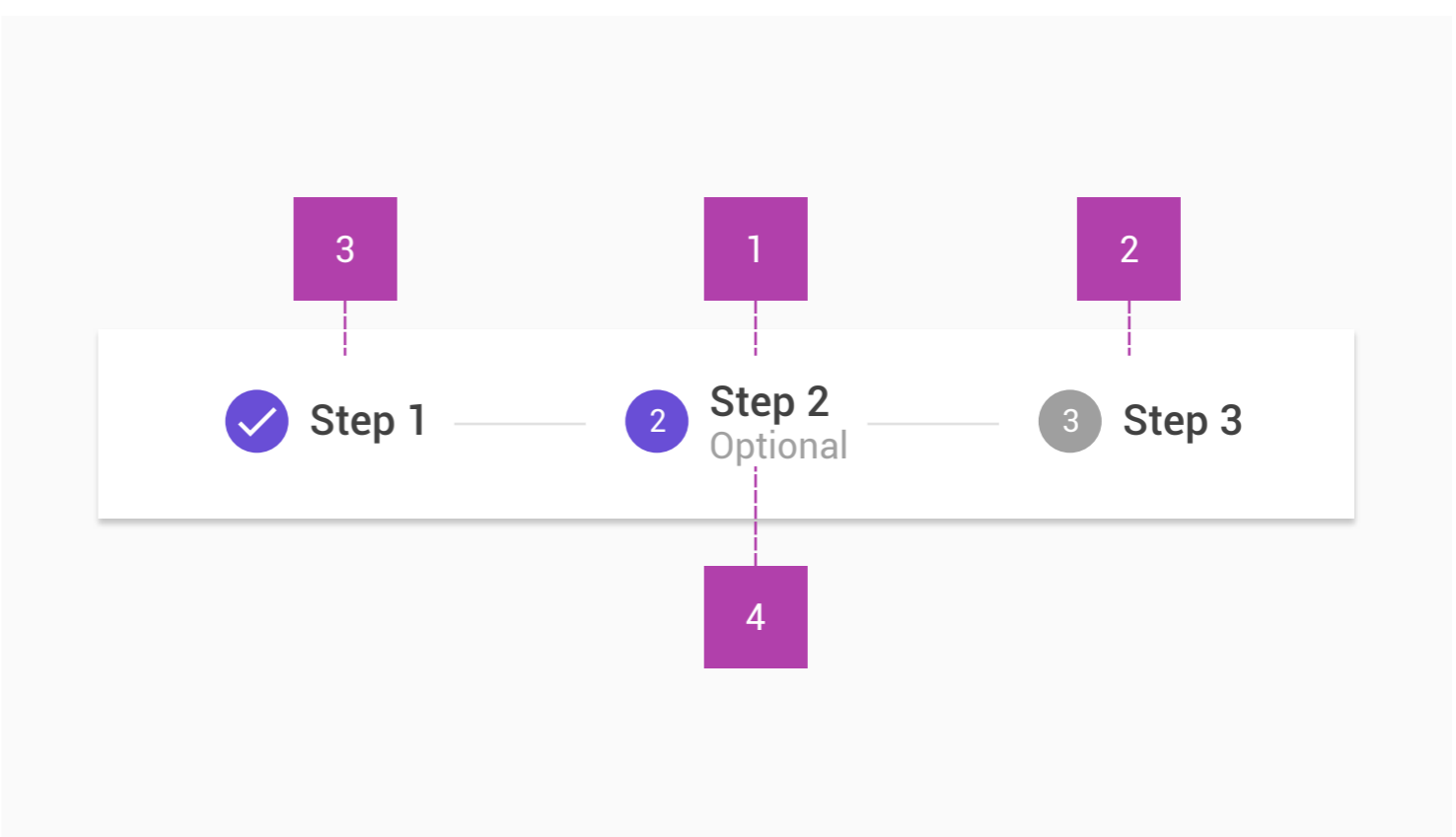
Steppers display progress through a sequence by breaking it up into multiple logical and numbered steps.

Avoid using steppers to break up sections in a short form, or multiple times on one page.

STRUCTURE

Steppers are made up of the following components:

- 1. Active step that indicates the current step that the user is on.
- 2. Inactive steps that indicate the steps that the user has yet to take action with.
- 3. Completed steps that indicate the steps that the user has already taken action with.
- 4. Optional label that indicates steps that are optional.



TYPES OF STEPS

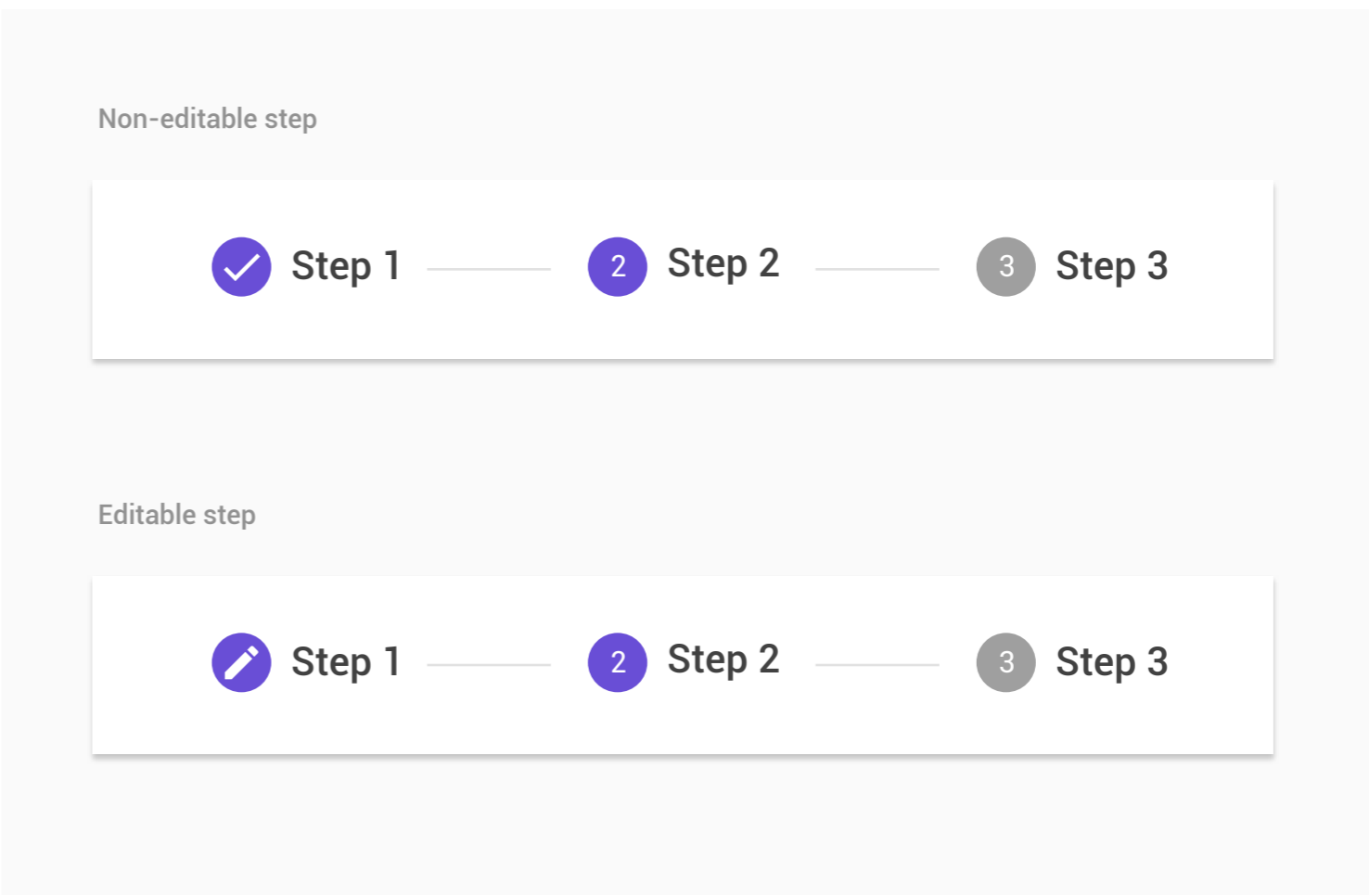
Editable steps

Editable steps allow users to return later to edit a step. These are ideal for workflows that involve editing steps within a session.

Non-editable steps

Non-editable steps should be used when:

- Users cannot edit a step later
- Step editing poses a distraction risk to form completion



TYPES OF STEPPERS

Horizontal steppers

Horizontal steppers are ideal when the contents of one step depend on an earlier step.

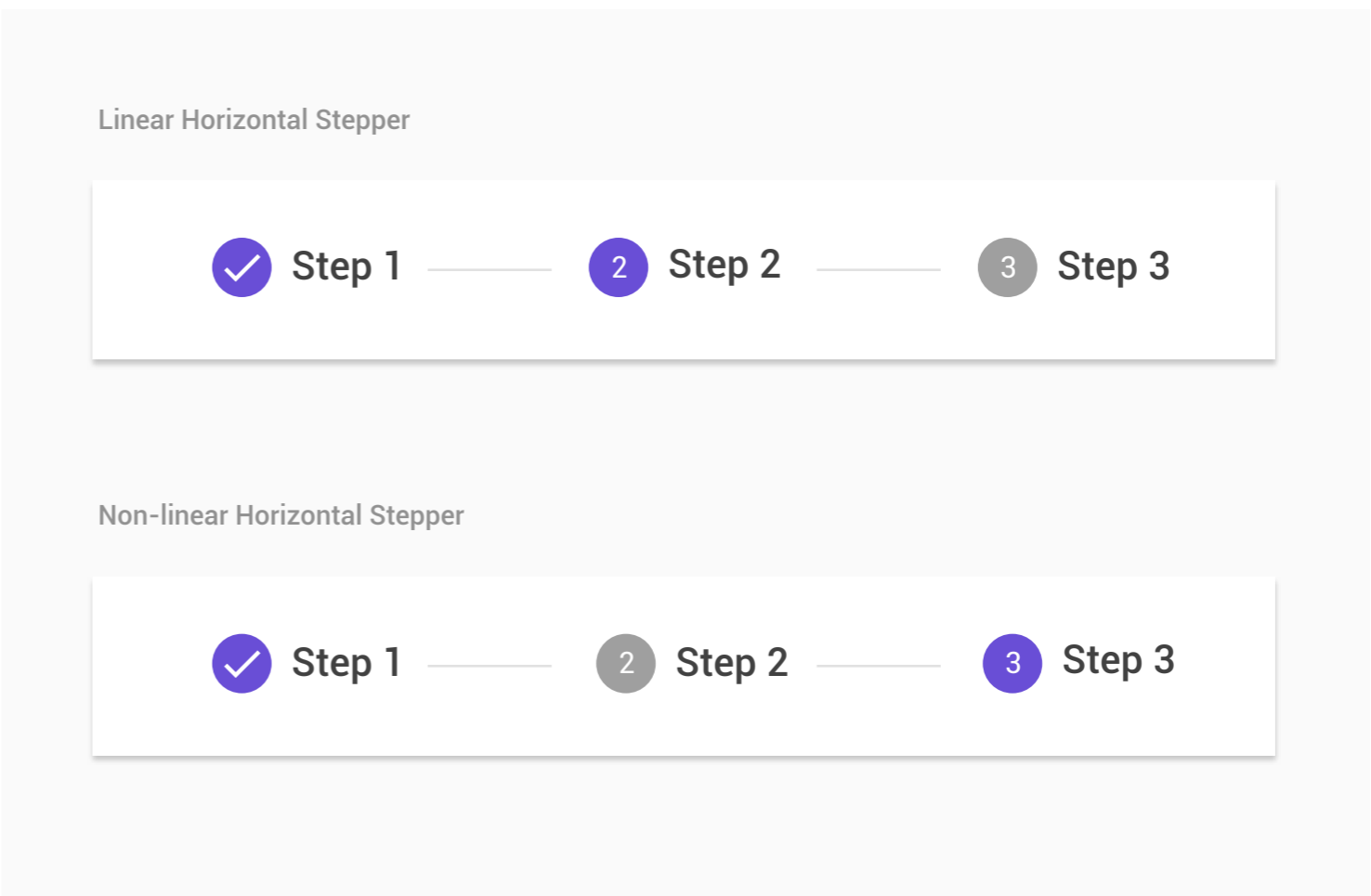
Avoid using long step names in horizontal step

Linear steppers

Linear steppers require users to complete one step in order to move on to the next.

Non-linear steppers

Non-linear steppers allow users to enter a multi-step flow at any point.



SPECIFICATIONS

Active stepper circle

Size: 24px x 24px

Text: 12px Roboto Regular

Color: V100

Active step

Text: 14sp Roboto Medium

Color: C100

Connector line

Size: 1px border

Color: C25

Inactive stepper circle

Size: 24px x 24px

Text: 12px Roboto Regular

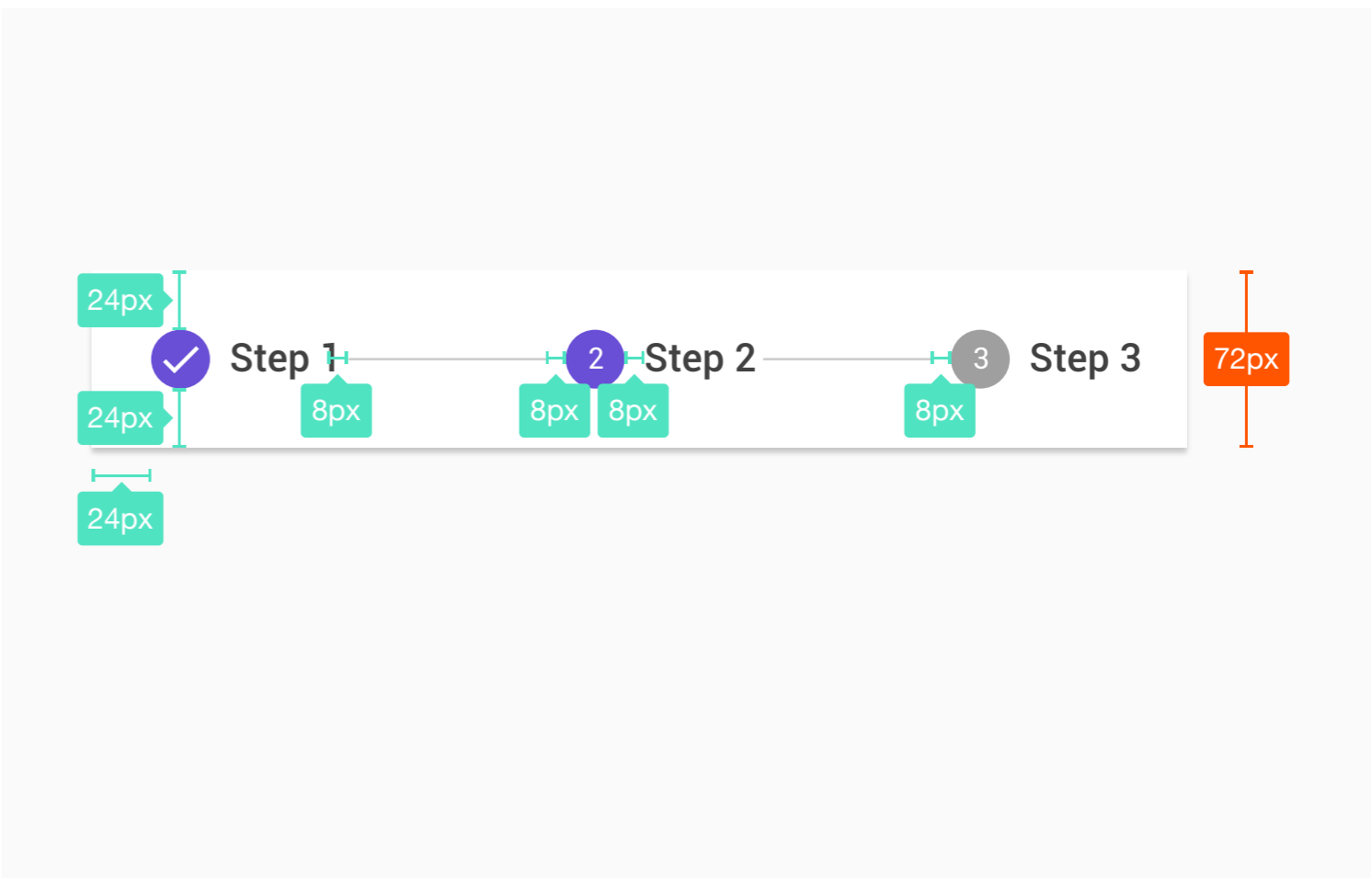
Text color: W100

Circle color: C25

Inactive steps

Text: 14px Roboto Regular

Color: C50



Standart Stepper

Step height: 72dp

Icon top, bottom, and left padding: 24dp

Icon right padding: 8dp

Optional step

Icon left and right padding: 8dp

Label right padding: 8dp

VARIATIONS

Alternative Label Position

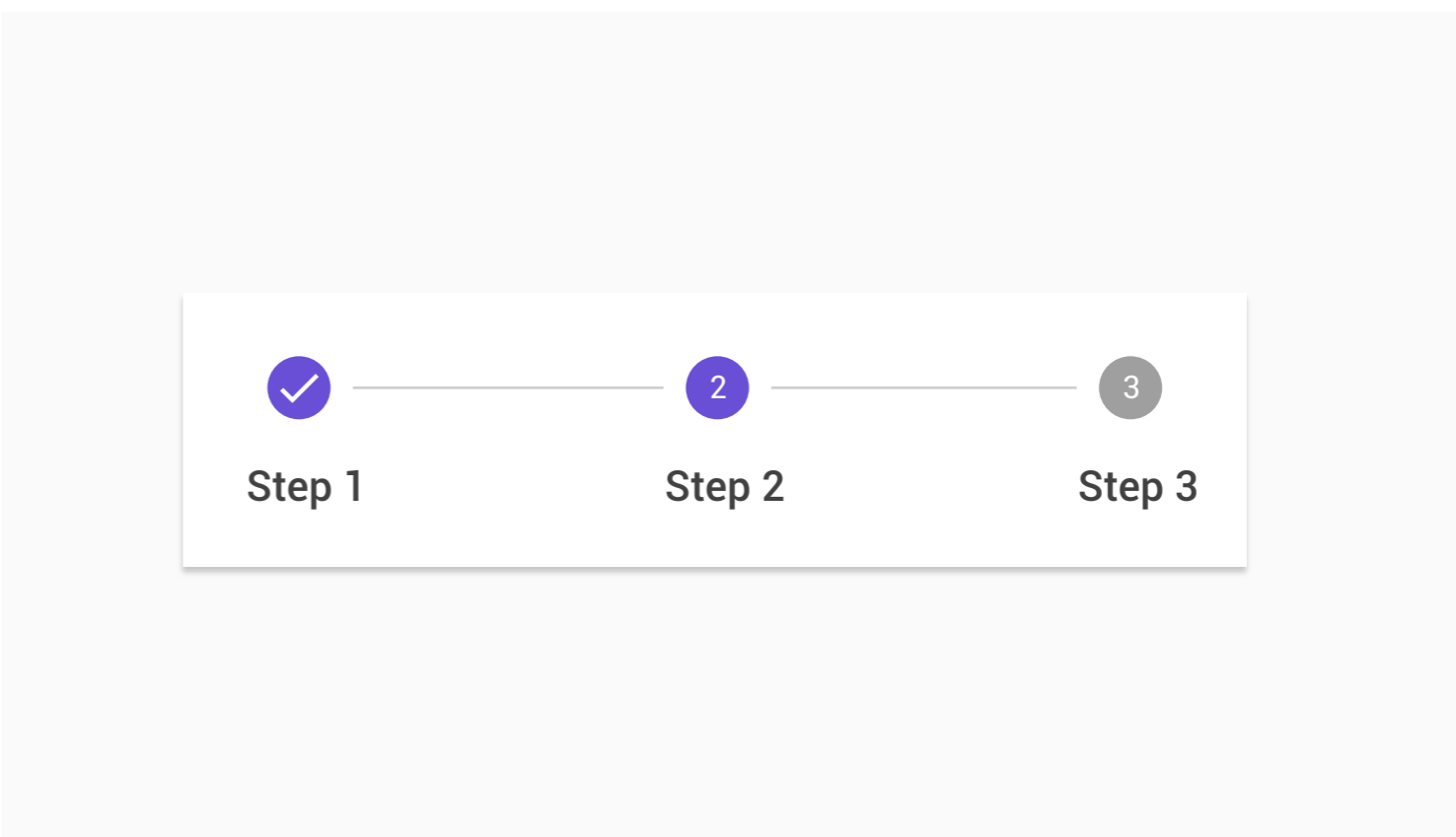
This is an alternative label placement, used when horizontal space may be limited.

Step height: 104dp

Icon left and right padding: 8dp

Label top padding: 16dp

Icon and label padding from edge: 24dp



REASONING

Why have we chosen to implement tool tips the way we've stated above.

SINGLE LINE NAVIGATION

TEMPLATE

The single line navigation is a variation of the Finastra headers and spans all other parts of the UI. The functionality in the header is applicable across all contexts in the Finastra ecosystem (internal or 3rd party). The following requirements need to be met for designing the Single line Header:

Global elements:

- Logo (Branding)
- Application Title (if applicable)
- User Account Access
- Notifications
- Elastic Search/Menu where applicable
- Menu (where Vertical menu is required)
- Global Applications Settings
- All elements (depending on the product usage) be visible on all viewports.
- Application "Return to Home" Button (this act as "Page Refresh" when home page is not set).
- Accessible (WCAG 2.0 and WAI-ARIA) - use of keyboard and voice.
- Palindromes (ability to be adjusted for right-to-left and left-to-right readability/usability).

Product-specific elements elements:

- Highest hierarchy level navigational URL's
- Dynamic Application Specific Actions (Favourites, Groups etc -TBC) per URL on hover.

USAGE - points to view/switch between navigational links where viewport does not allow visibility of all (overflow).

Groups & Settings Elements

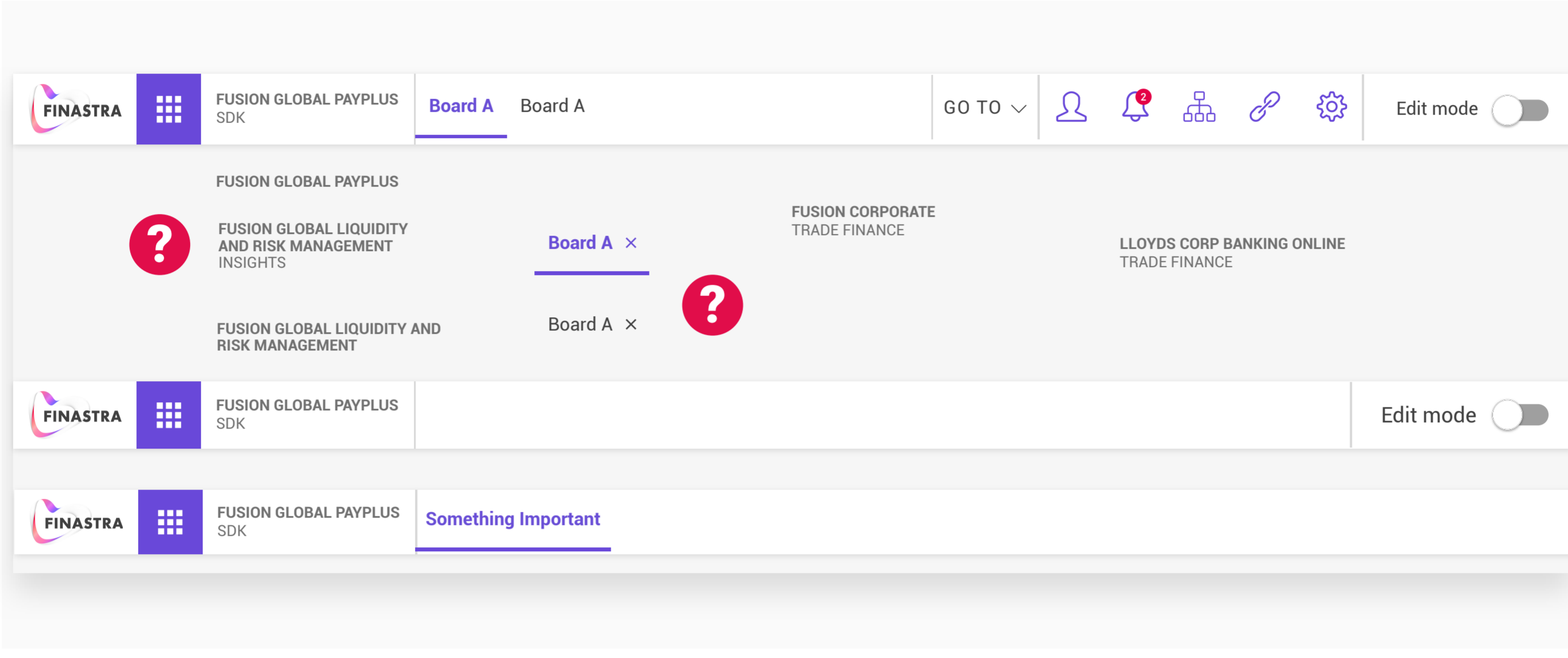
Single line horizontal navigation should be used in products where the double line nav is not applicable due to reasons such as limited screen real estate.

When creating boards for each workspace, try to avoid creating too many boards and forcing the user to use the "Go To" dropdown. Minimise use of the dropdown and hidden navigation items.

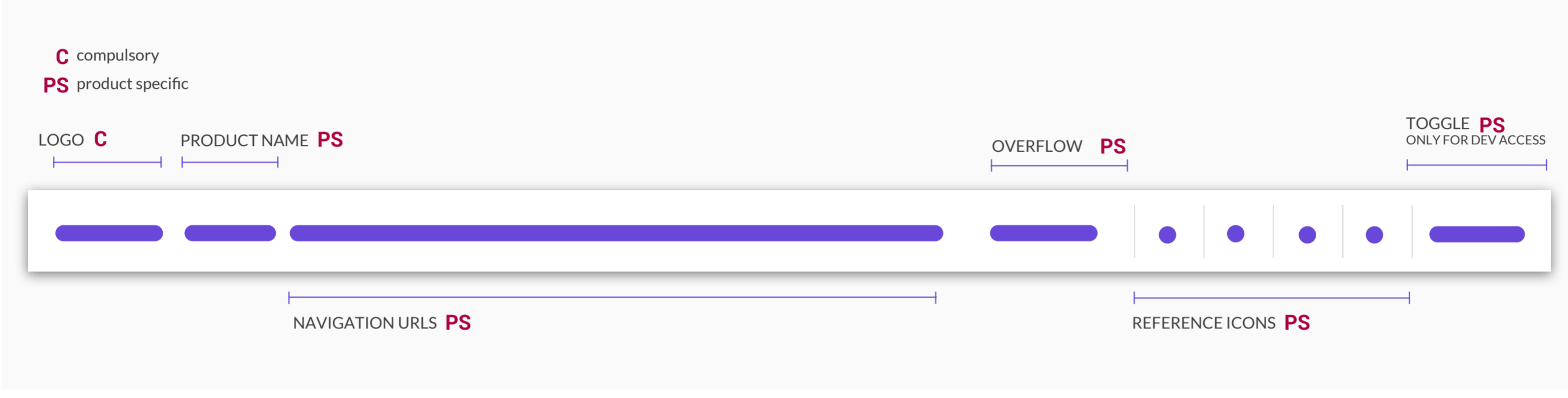
Clicking on the Finastra logo should take users back to the homepage.

Additional do's and don't's...

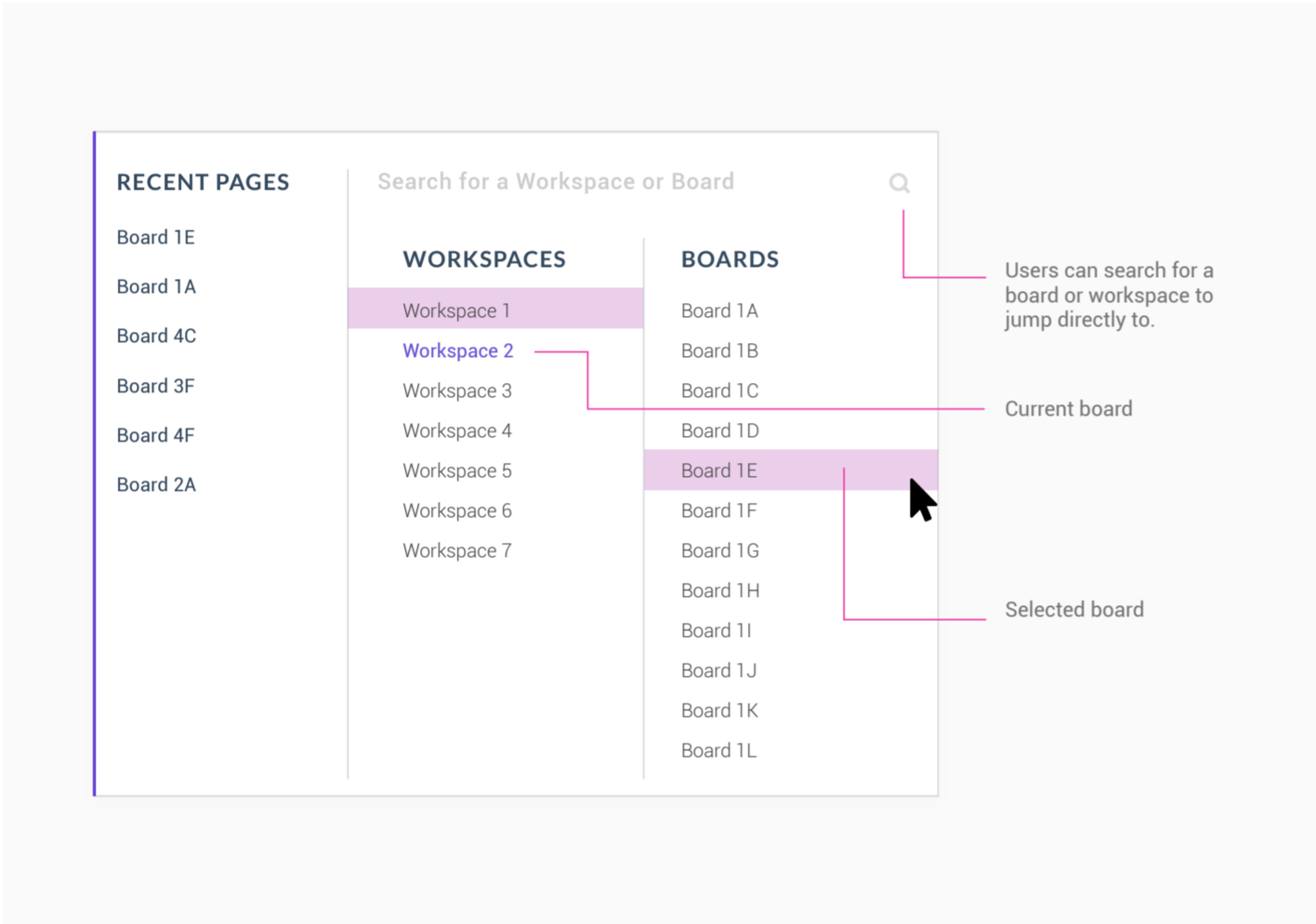
STRUCTURE



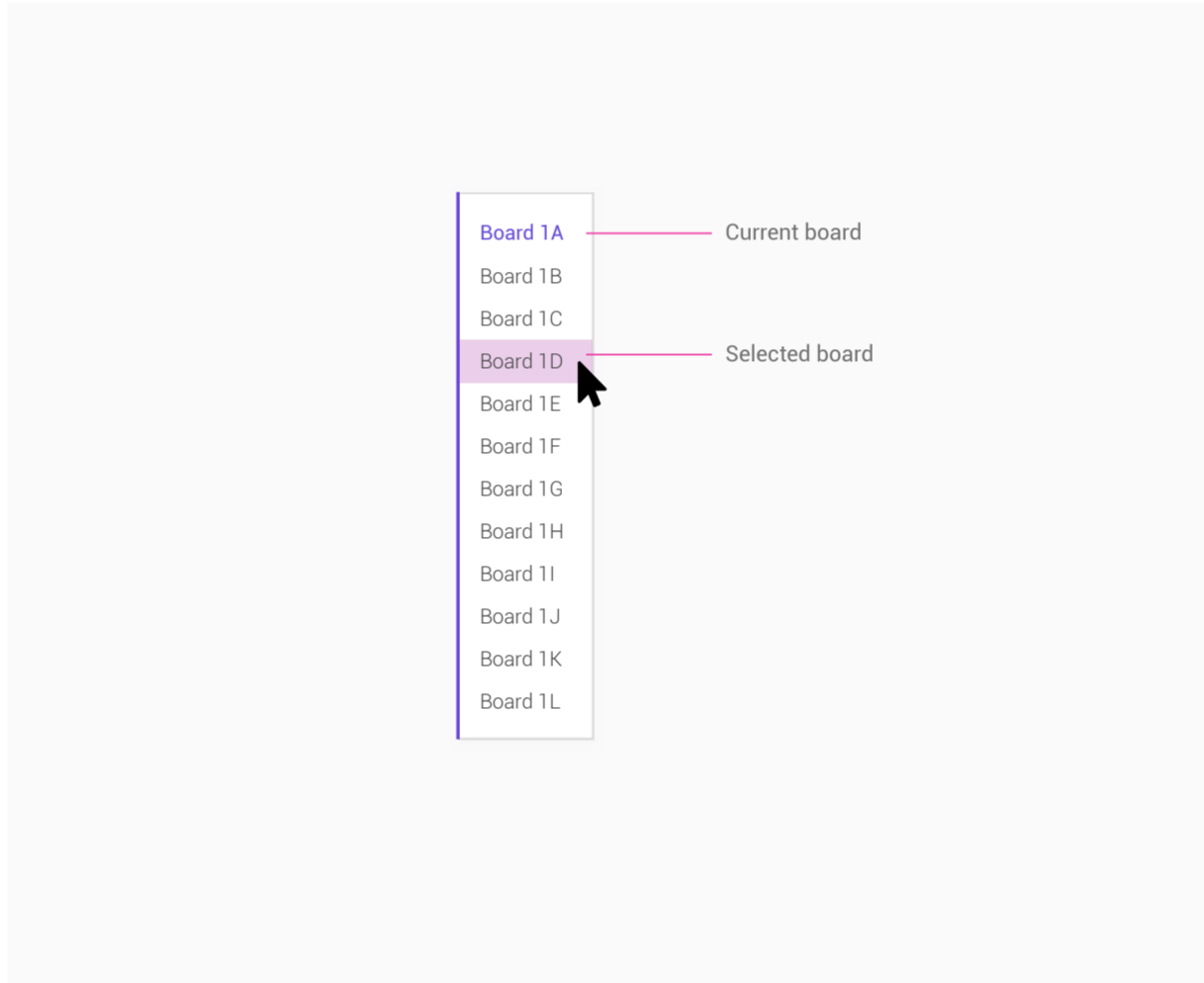
The single line header includes the following components:



The main menu dropdown looks as follows:



The Go To dropdown includes all boards for that workspace and looks as follows:

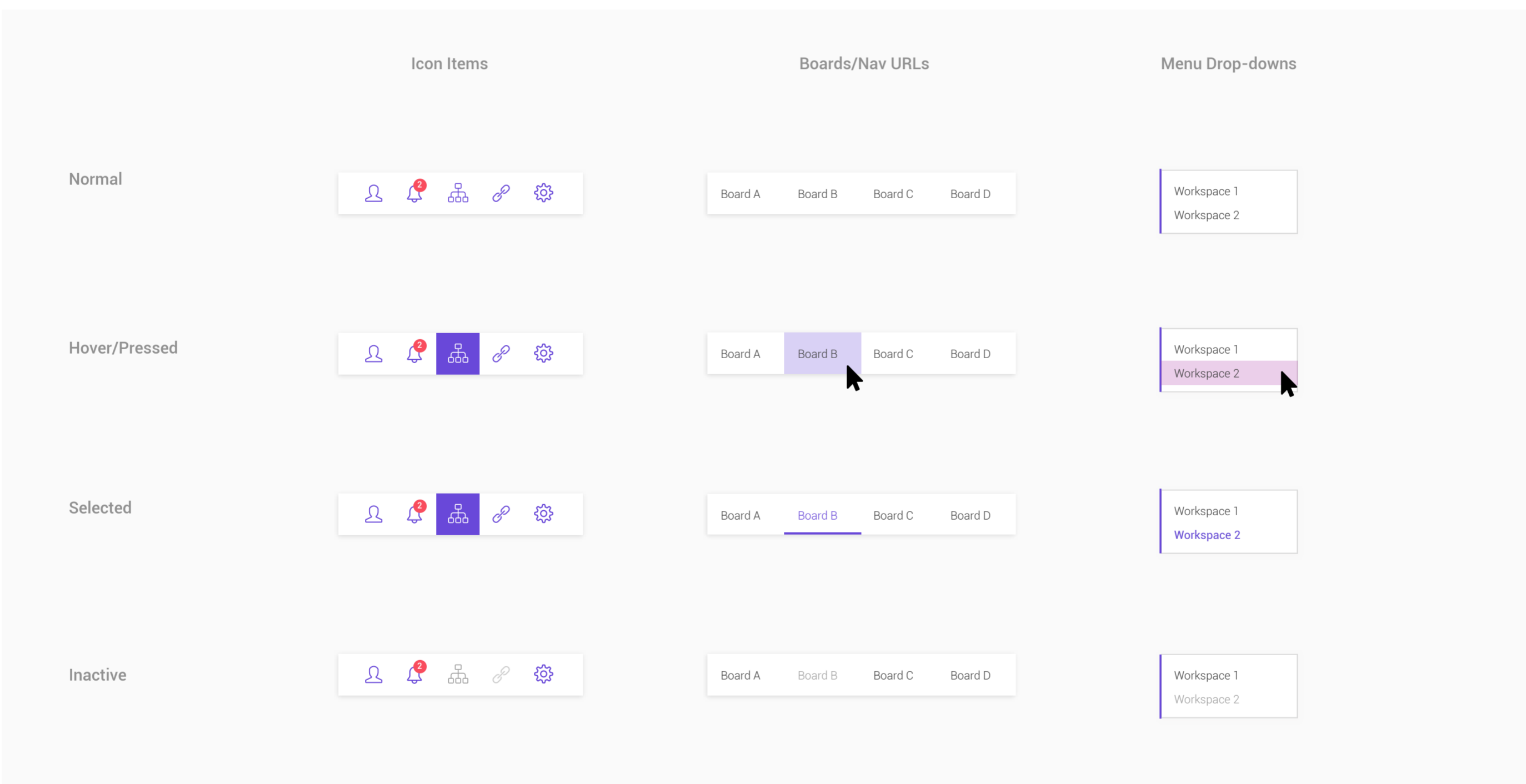


STATES

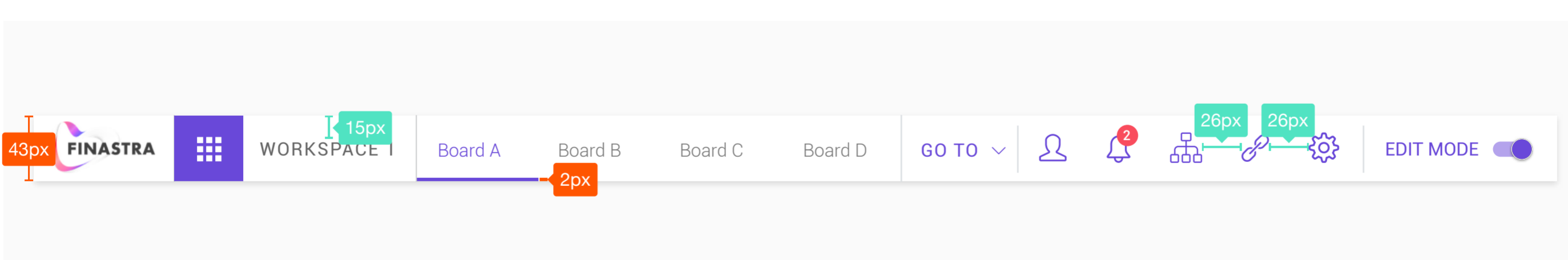
Disclaimer: the states and details of the navigation are specific to FUXP.

Most components in the navigation bar have 4 possible states:

- Normal
- Hover/Focused/Pressed
- Inactive



SPECIFICATIONS



State	Icons	Boards	Drop-downs
Normal/unselected	Fill Color: V100 Background color: transparent	Text: Roboto Light 12px Text Color: C100 Background color: transparent	Text: Roboto Light 12px Text Color: C100 Background color: transparent
Hover	Fill Color: W100 Background color: V100	Text: Roboto Light 12px Text Color: C100 Background color: V25	Text: Roboto Light 12px Text Color: C100 Background color: F25
Selected	Fill Color: W100 Background color: V100	Text: Roboto Light 12px Text Color: V100 Background color: Transparent Stroke color: V100	Text: Roboto Regular 12px Text Color: V100 Background color: transparent
Disabled	Fill Color: C50 Background color: transparent	Text: Roboto Light 12px Text Color: C50 Background color: transparent	Text: Roboto Light 12px Text Color: C50 Background color: transparent

VARIATIONS

Here are some examples of exceptions and variations to the guidelines above.

REASONING

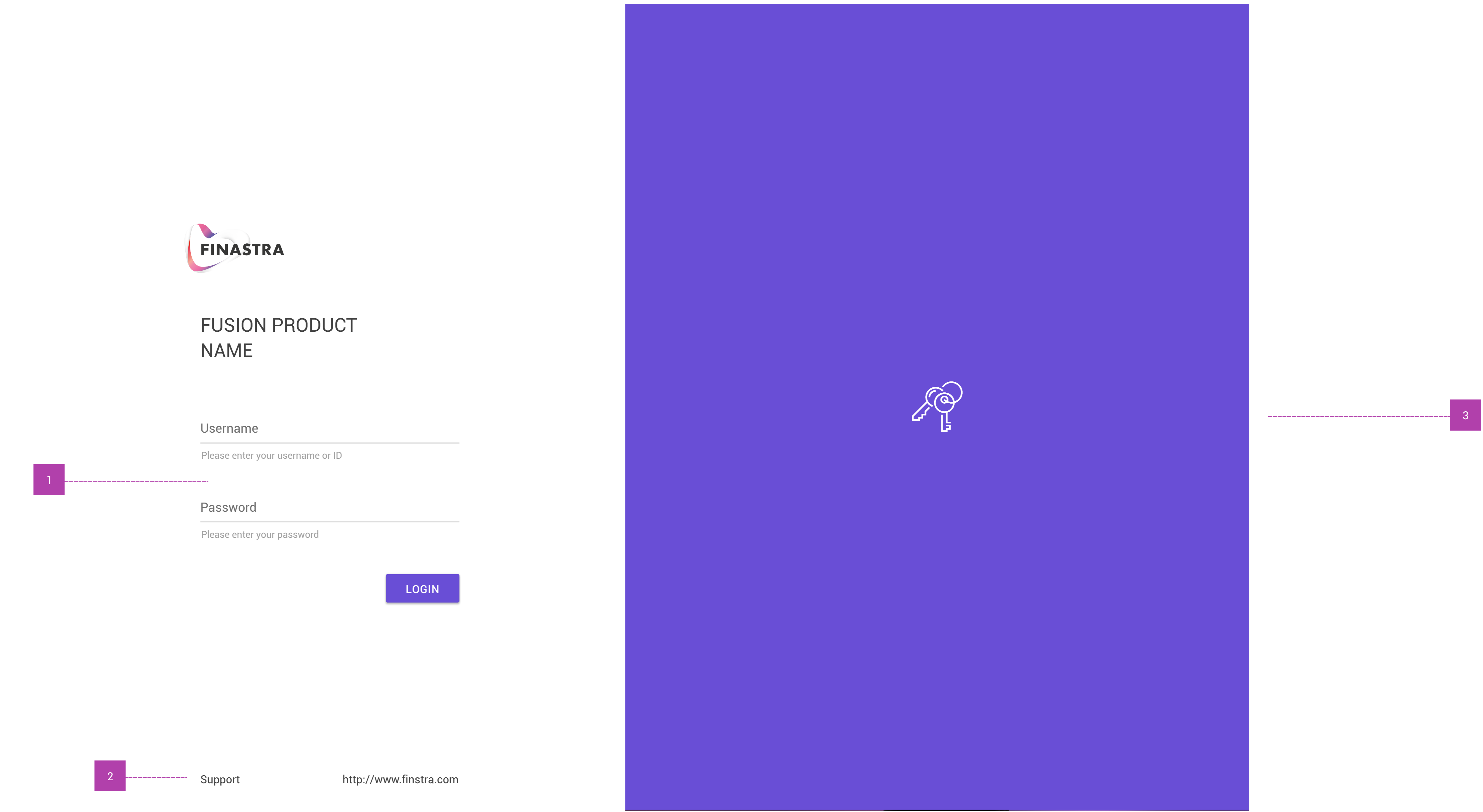
Why have we chosen to implement tool tips the way we've stated above.

LOGIN SCREEN

The Login Screen represents the gateway and staring point access of our products. Fused with brand alignment it enables customisation and flexibility.

LOGIN 50/50 Version

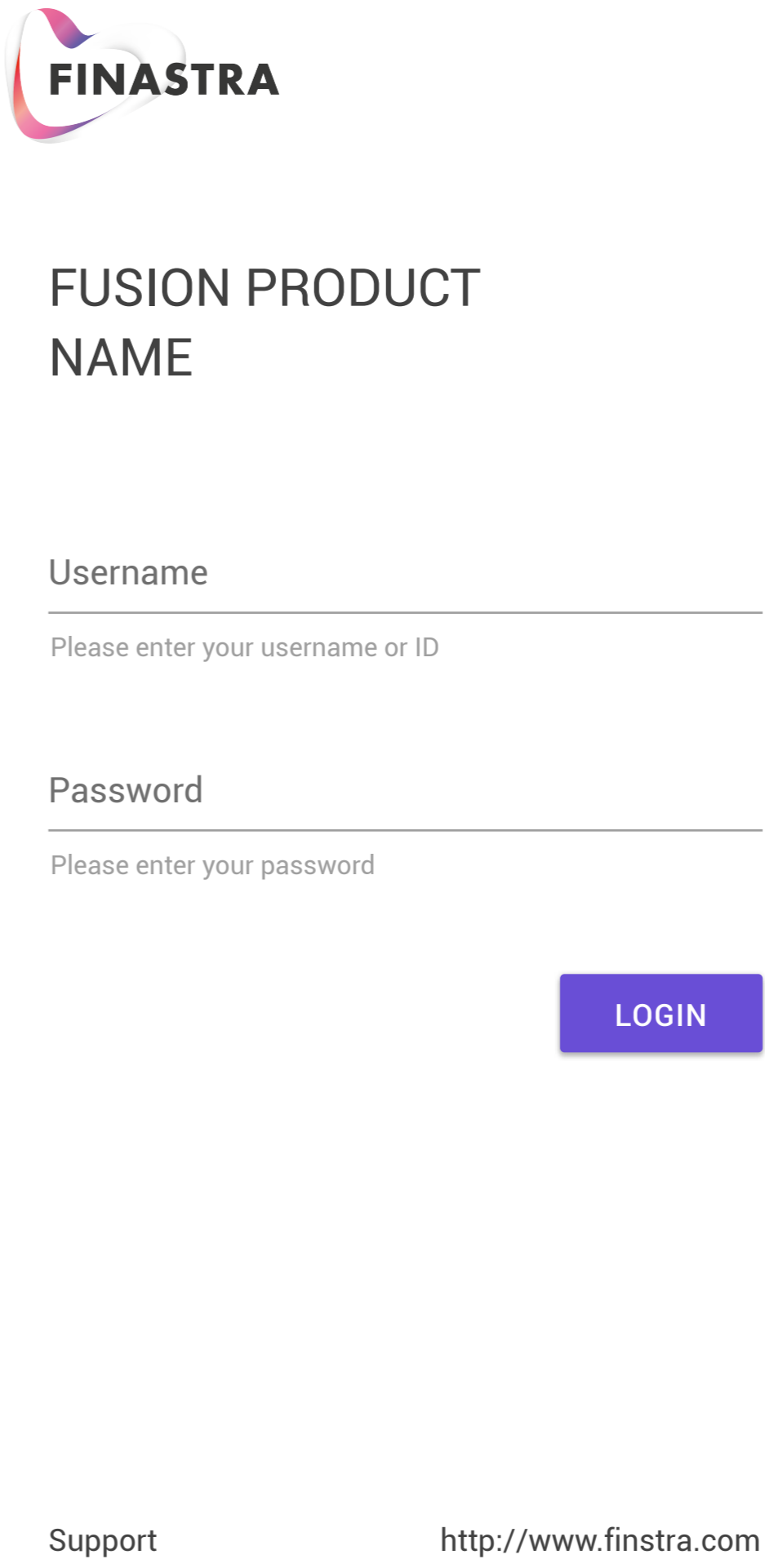
The 50/50 version enables both LTR and RTL options.



- Text fields contain the following elements:
- 1. Lgoin Group: Input fields labels display the type of in put a field requires and the CTA for Login.
 - 2. Hyperlinks: Support and Finastra.com direction hyperlinks.
 - 3. Login Iconr: Default Icon for template based login screen.

LOGIN 3/1 Ratio + Brand Fusion

The 3/1 ratio version enables both LTR and RTL options including branded fusion imagery.



TIMELINE - Pattern Definition & Design

Definition: Lays out the steps in a process sequentially, from start to end, so that the user can see a clear sequence of what needs to be done to get through a path. The timeline gives the user a grasp on an otherwise ambiguous process with a clear, glanceable representation of a complex series of steps.

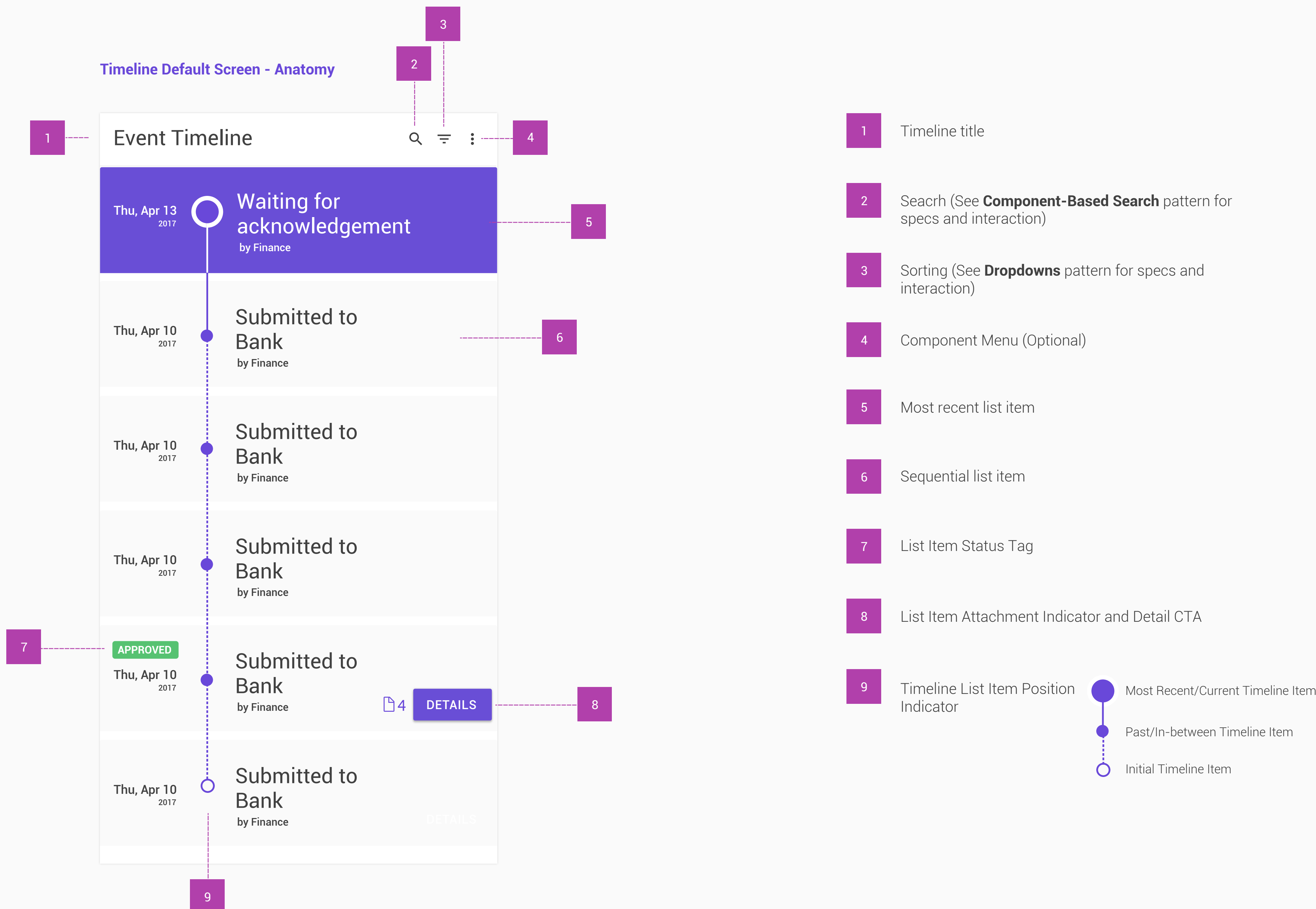
A timeline can:

- record units in milestones
- cannot show the future past current date
- is sequential

Related Pattern Definition

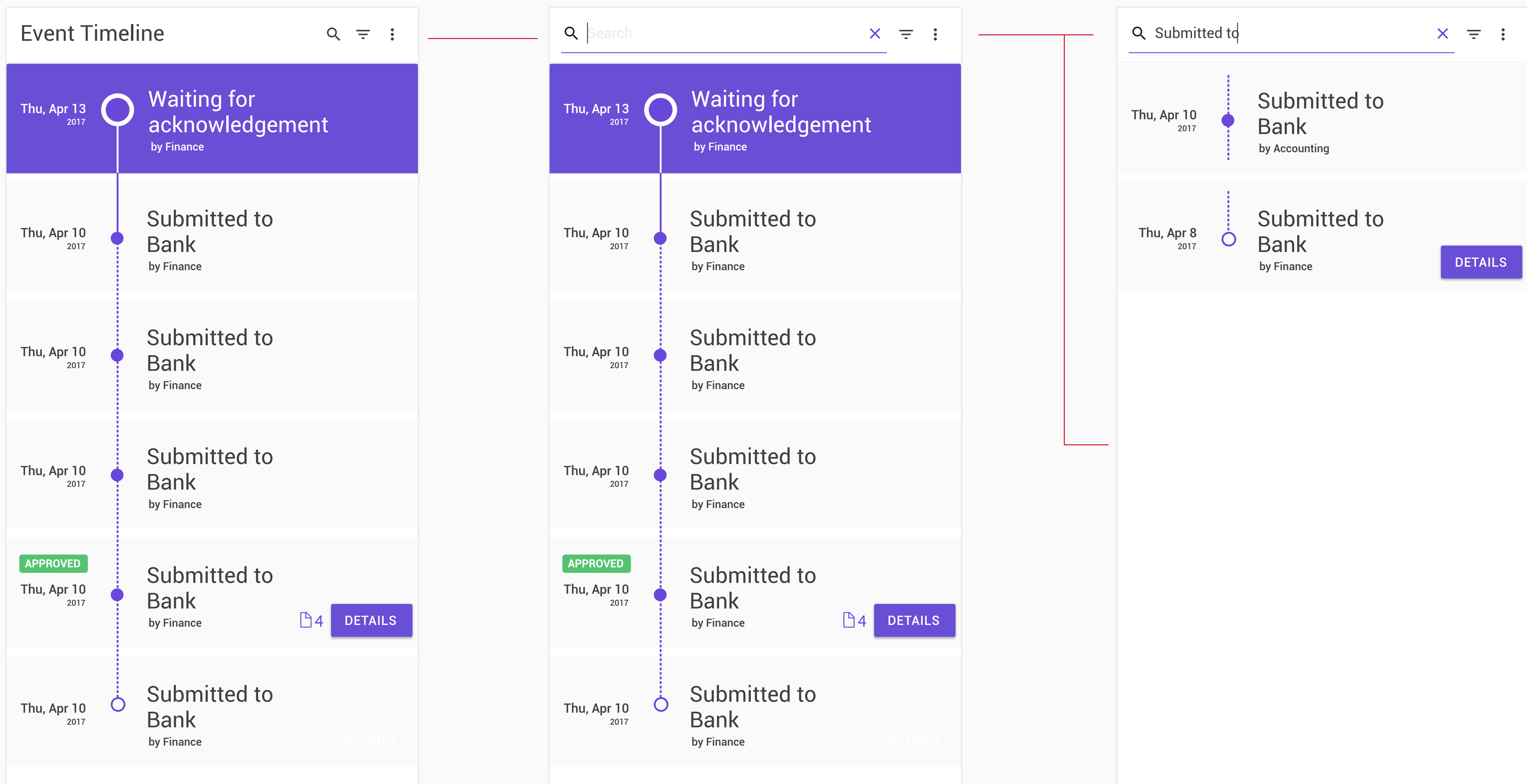
STEPPER

- contains milestones
- is also sequesntial but to highest level

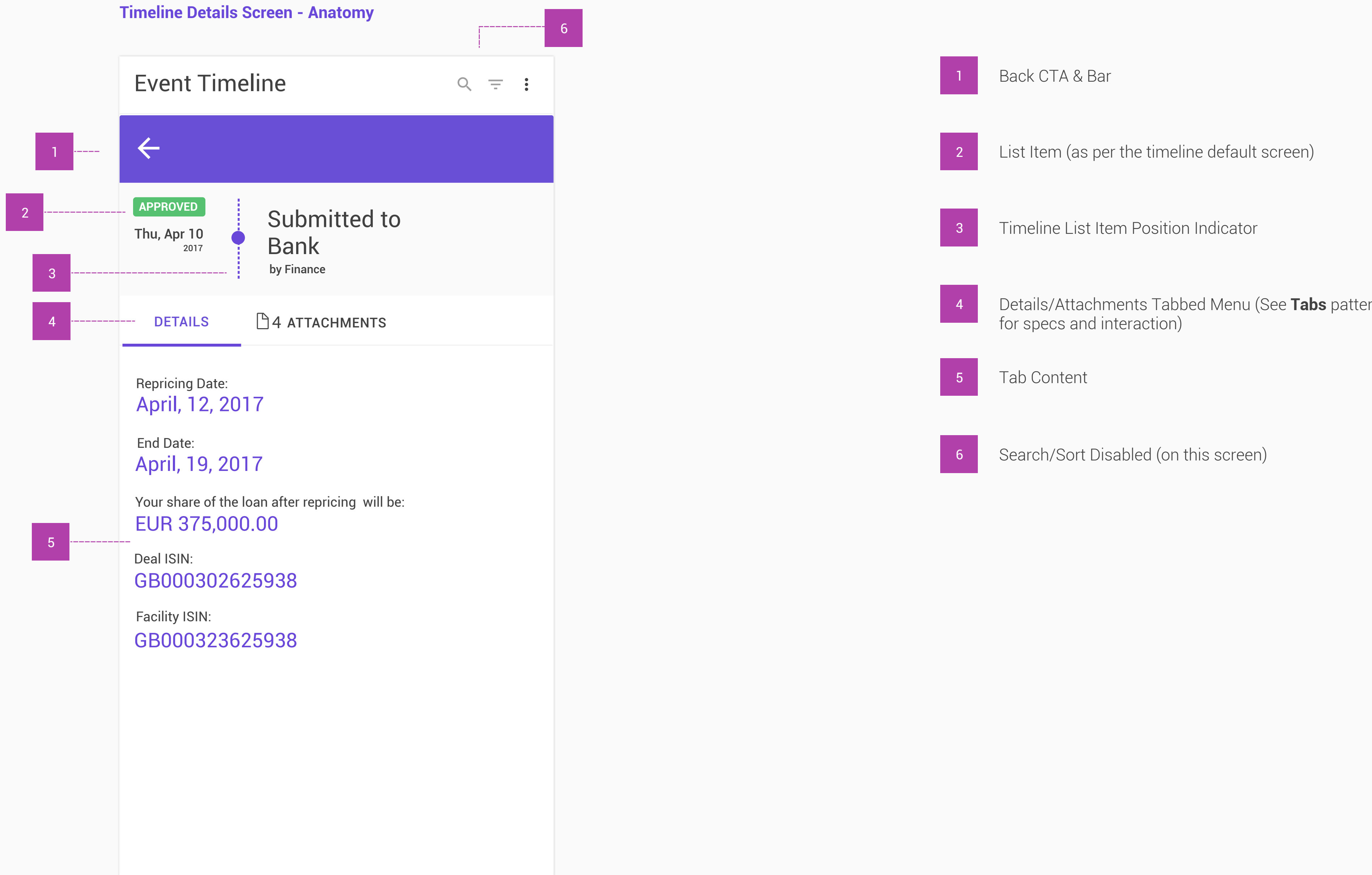


Timeline Default Screen - Interactions

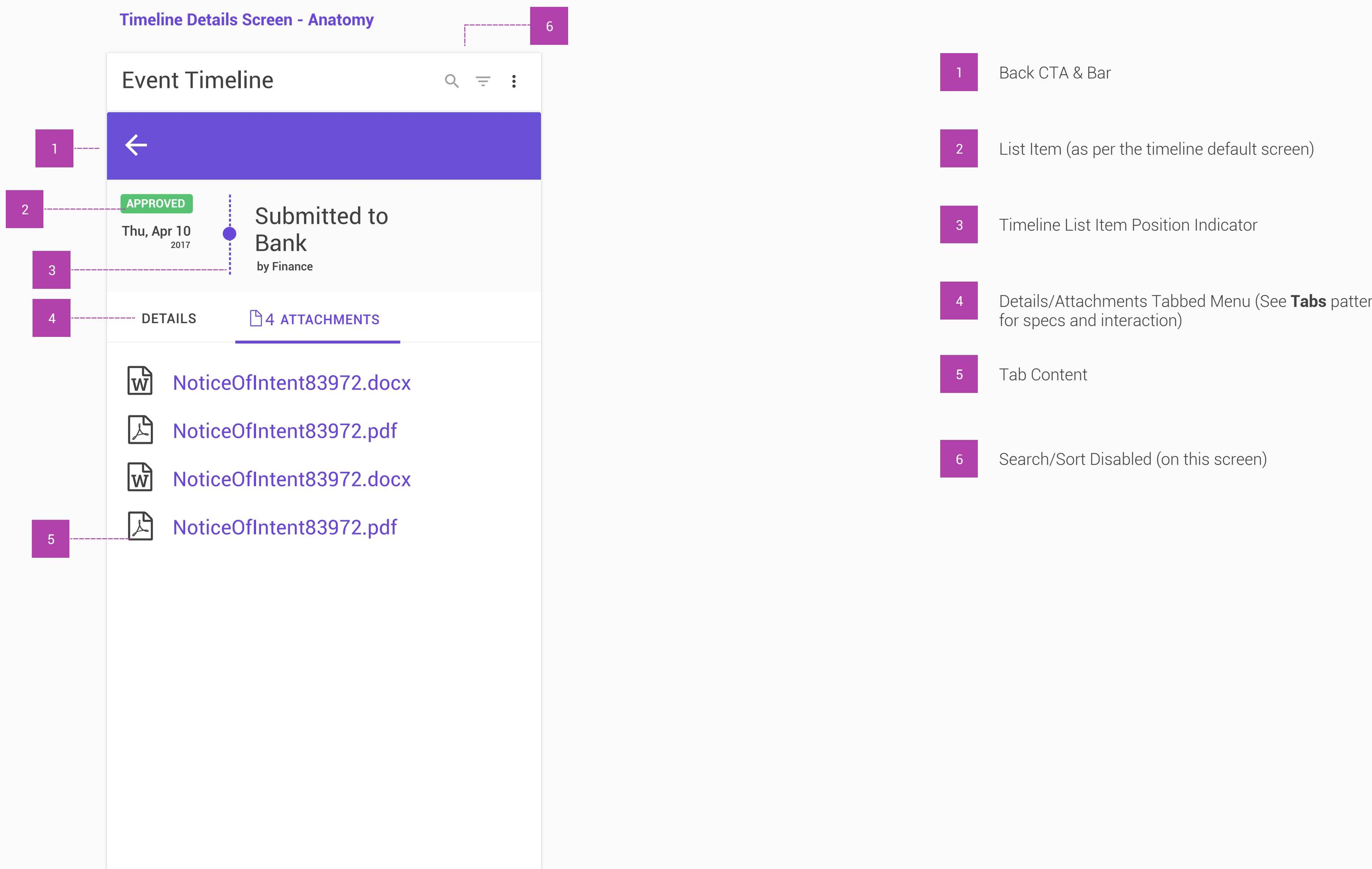
SEARCH



Timeline Details Screen - Anatomy



Timeline Details Screen - Anatomy



10.0 SLIDERS

Sliders let users select from a range of values by moving the slider thumb.

Sliders are ideal components for adjusting settings that reflect intensity levels, such as volume, brightness, or color saturation.

Sliders may have icons on both ends of the bar that reflect the value intensity. Place the smallest value for the slider range on the left and the largest value on the right.

Continuous sliders

Continuous sliders allow users to select a value along a subjective range. They do not require a specific value to make adjustments, although they may, in some instances, offer an editable numeric value.

Discrete sliders

Discrete sliders allow users to select a specific value from a range.

10.1 CONTINUOUS SLIDER

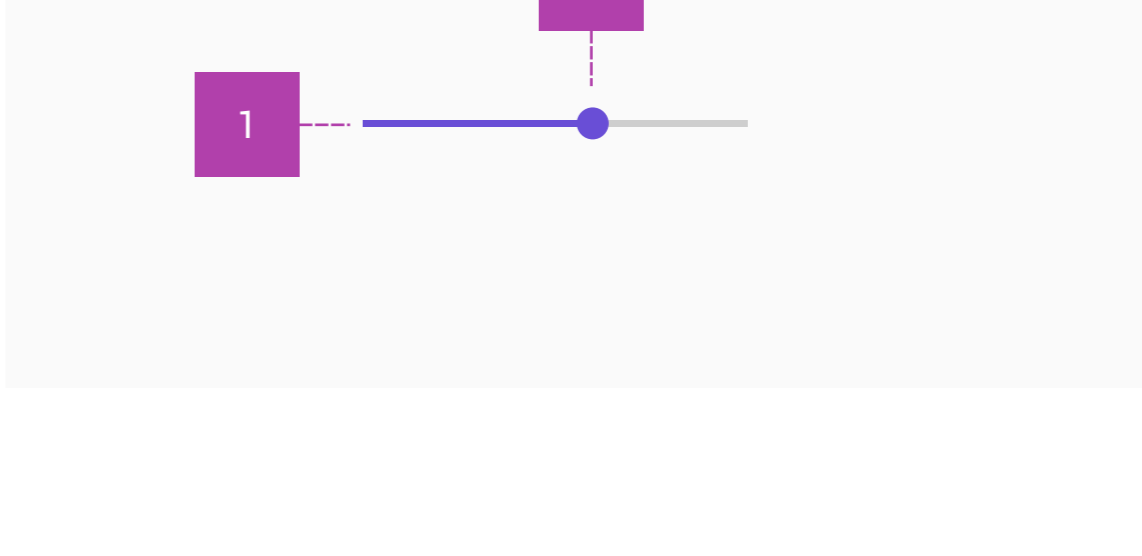
USAGE

Use continuous sliders for subjective settings that do not require a specific value for the user to make meaningful adjustments.

STRUCTURE

The switch is comprised of 2 pieces:

1. The track, which indicates the range of values
2. The thumb, the circle: The position fo the thumb on the track indicates its value.



STATES

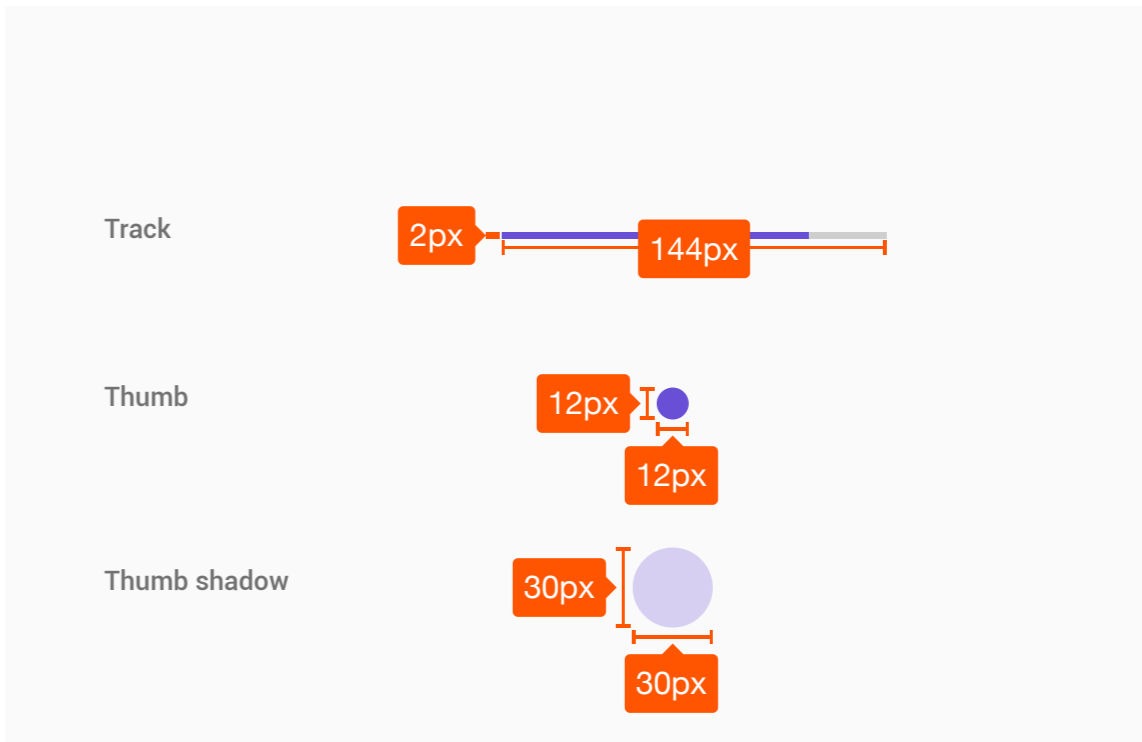
Checkboxes have 3 major states: enabled or disabled.

In the enabled state, the following user interactions are available:

- Idle/default
- Hover
- Click



SPECIFICATIONS



State	0 Value	> 0 Value	
Normal/unselected	Border: C25, 1px Color: W100 Elevation: 2px	Type: Roboto 16px Color: C100	Menu item height: 32px Color: C10

10.2 DISCRETE SLIDER

The discrete slider thumb snaps to evenly spaced tick marks along the slider rail. Use for objective settings that require specific values for the user to make meaningful adjustments. Each tick mark should change the setting to a level that's discernible to the user. The values are predetermined and aren't user-editable.

USAGE

Use for settings for which users need to know the exact value of the setting.

STRUCTURE

The switch is comprised of 2 pieces:

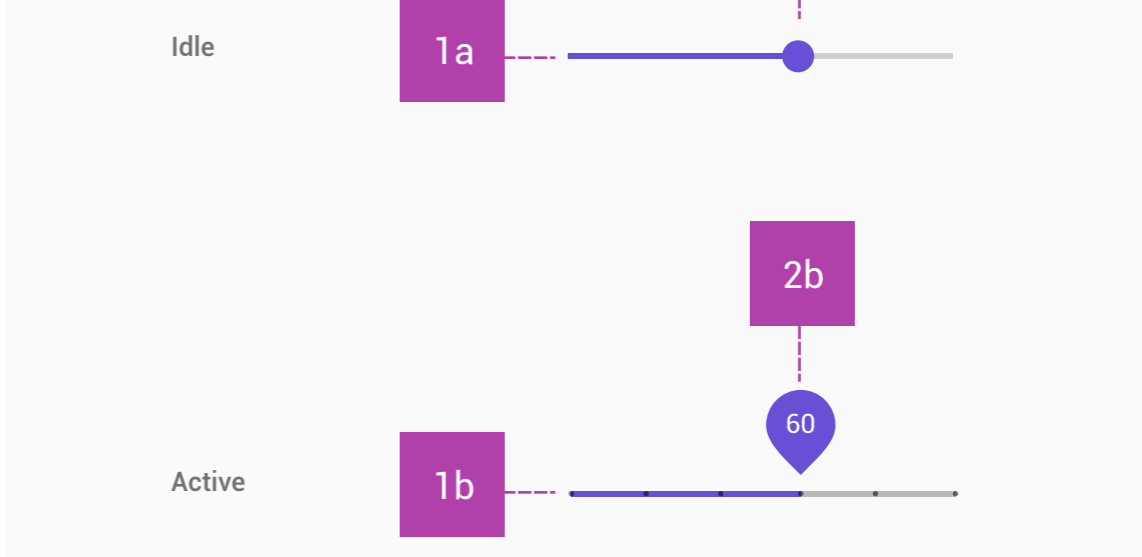
1. The track, which indicates the range of values. For the discrete slider, this has 2 major states:
 - a. Idle: when the user is not interacting with it
 - b. Active: the slider track has visible tick marks the reflect predetermined value increments.
2. The thumb, the circle: The position of the thumb on the track indicates its value. For the discrete slider, this has 2 major states:
 - a. Idle: when the user is not interacting with it
 - b. Active: the thumb turns into an indicator displaying the exact value.

STATES

Checkboxes have 3 major states: enabled or disabled.

In the enabled state, the following user interactions are available:

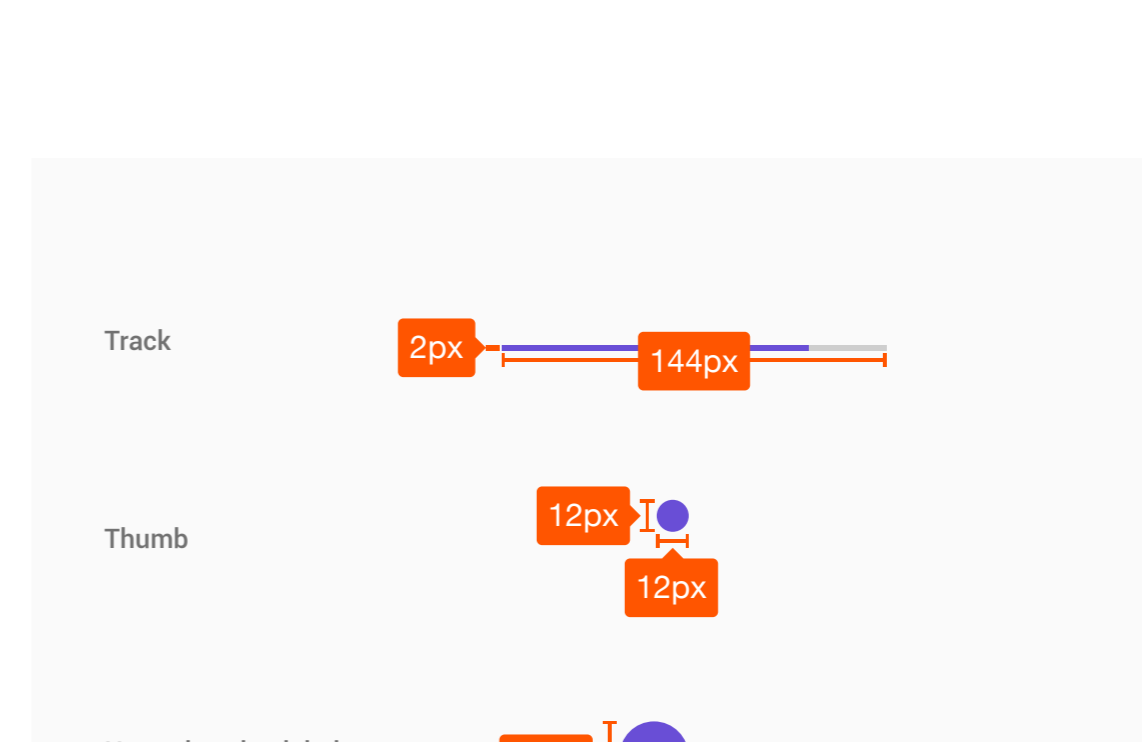
- Idle/default
- Hover
- Click



SPECIFICATIONS

Numeric value text: Roboto regular 10px

Numeric value text color: W100



State	0 Value	> 0 Value
Normal	Base Track Color: C25 Thumb: 12px x 12px Thumb fill color: C100	Base Track Color: C25 Filled track color: V100 Thumb: 12px x 12px Thumb fill color: V100

Hover	Base Track Color: C50 Thumb fill color: C50	Base Track Color: C50 Filled track color: V100 Thumb fill color: V100
-------	--	---

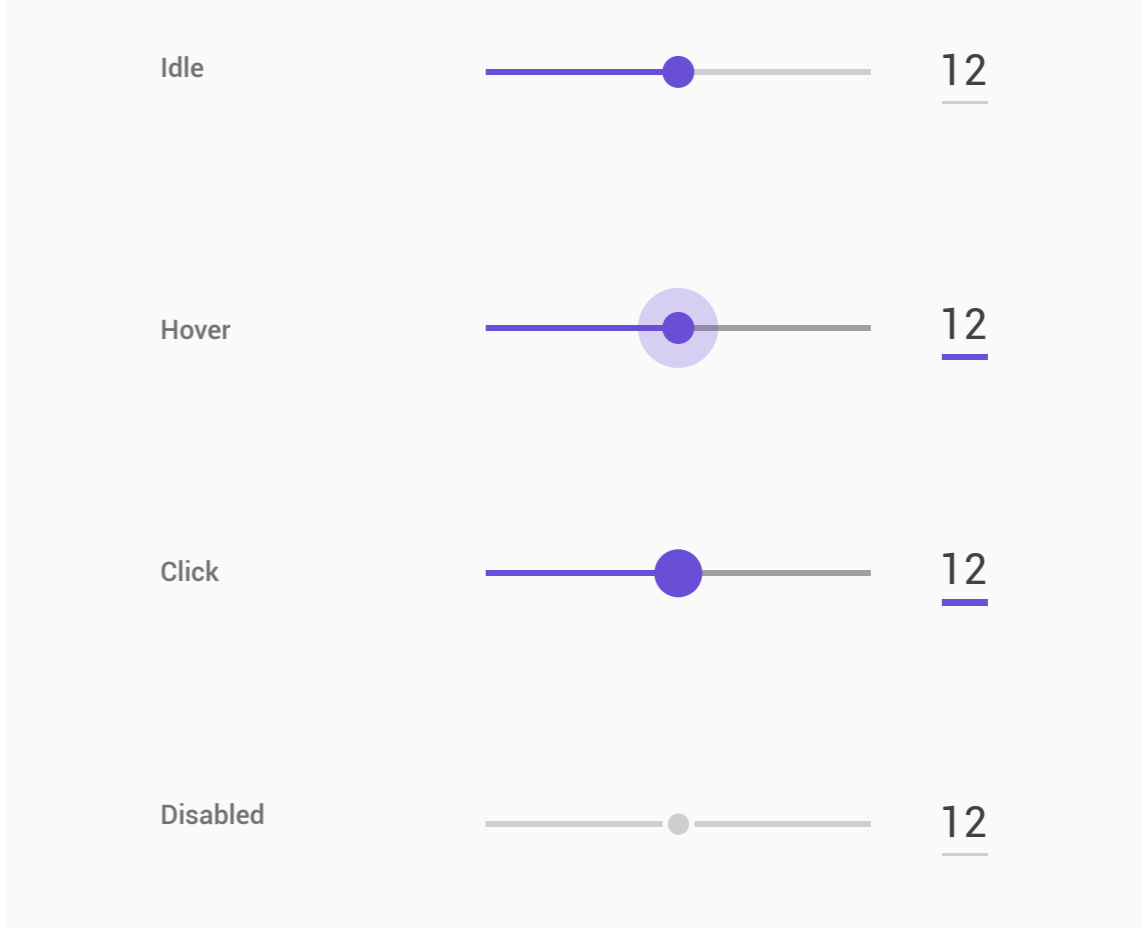
Click	Base Track Color: C50 Thumb fill color: C50 Track tick color: Black, #000000	Base Track Color: C50 Filled track color: V100 Thumb fill color: V100 Track tick color: Black, #000000
-------	--	---

Disabled	Base Track Color: C25 Thumb: 10px x 10px Thumb fill color: C25 Thumb stroke color: Transparent/background Thumb stroke: 2px	Base Track Color: C25 Filled track color: C25 Thumb: 10px x 10px Thumb fill color: C25 Thumb stroke color: Transparent/background Thumb stroke: 2px
----------	---	--

10.3 VARIATIONS

CONTINUOUS SLIDER WITH EDITABLE NUMERIC VALUE

Use for settings for which users need to set the exact value. Upon pressing the thumb, the text box becomes editable for text entry and updates the value automatically with thumb movement.



10.4 REASONING

Why have we chosen to implement tool tips the way we've stated above.

13.0 MATERIAL SNACKBAR NOTIFICATIONS

Snackbars provide brief feedback about an operation through a message at the bottom of the screen.

Snackbars contain a single line of text directly related to the operation performed. They may contain a text action, but no icons.

USAGE

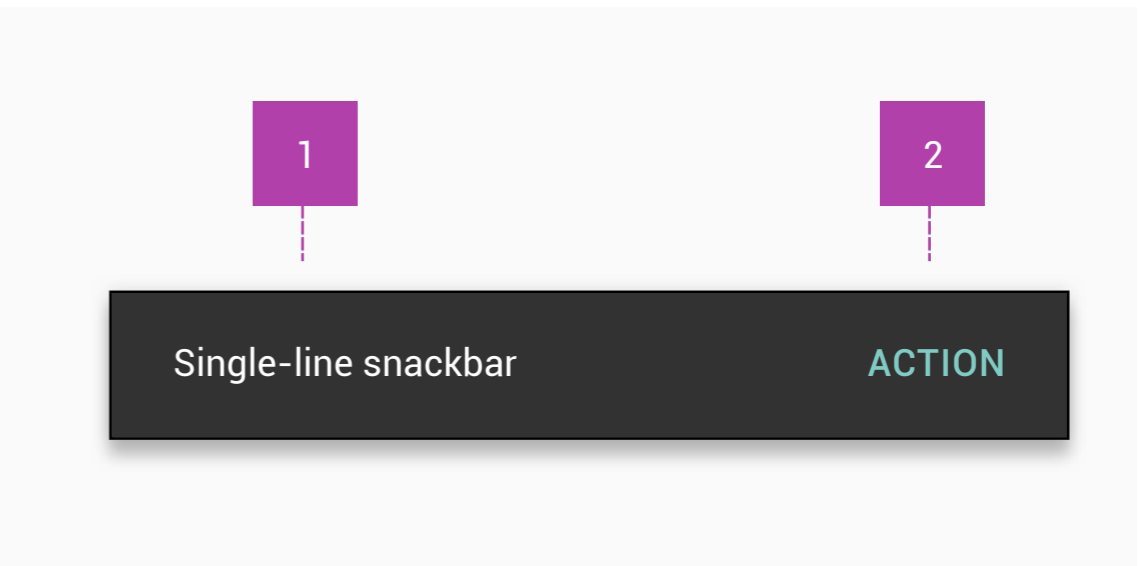
Only one snackbar may be displayed at a time. Each snackbar may contain a single action, neither of which may be “Dismiss” or “Cancel.”

13.1 STRUCTURE

Snackbars should only be made up of a maximum of two atoms:

1. The single line of text, which directly related to the operation performed.
2. OPTIONAL: The text action, neither of which may be “Dismiss” or “Cancel.”

If an action is present, comply with dialog spacing and affordance rules. For two or more actions, use a dialog, not a snackbar, even when one of the actions is a dismiss action. If the action described in the snackbar is important enough to block the use of the screen, it should be a dialog.



BEHAVIOUR

Upon entrance, snackbars animate upwards from the bottom edge of the screen. When they appear, they do not block input.

They exit by being swiped off-screen or automatically disappear after a timeout or user interaction elsewhere (such as summoning a new surface or activity).

Snackbars automatically time out from the screen. For usability reasons, snackbars should not contain the only way to access a core use case. They should not be persistent or be stacked, as they are above other elements on screen.

13.2 SPECIFICATIONS

Container

Single-line snackbar height: 48px tall

Minimum width: 288px

Maximum width: 568px

2px rounded corner

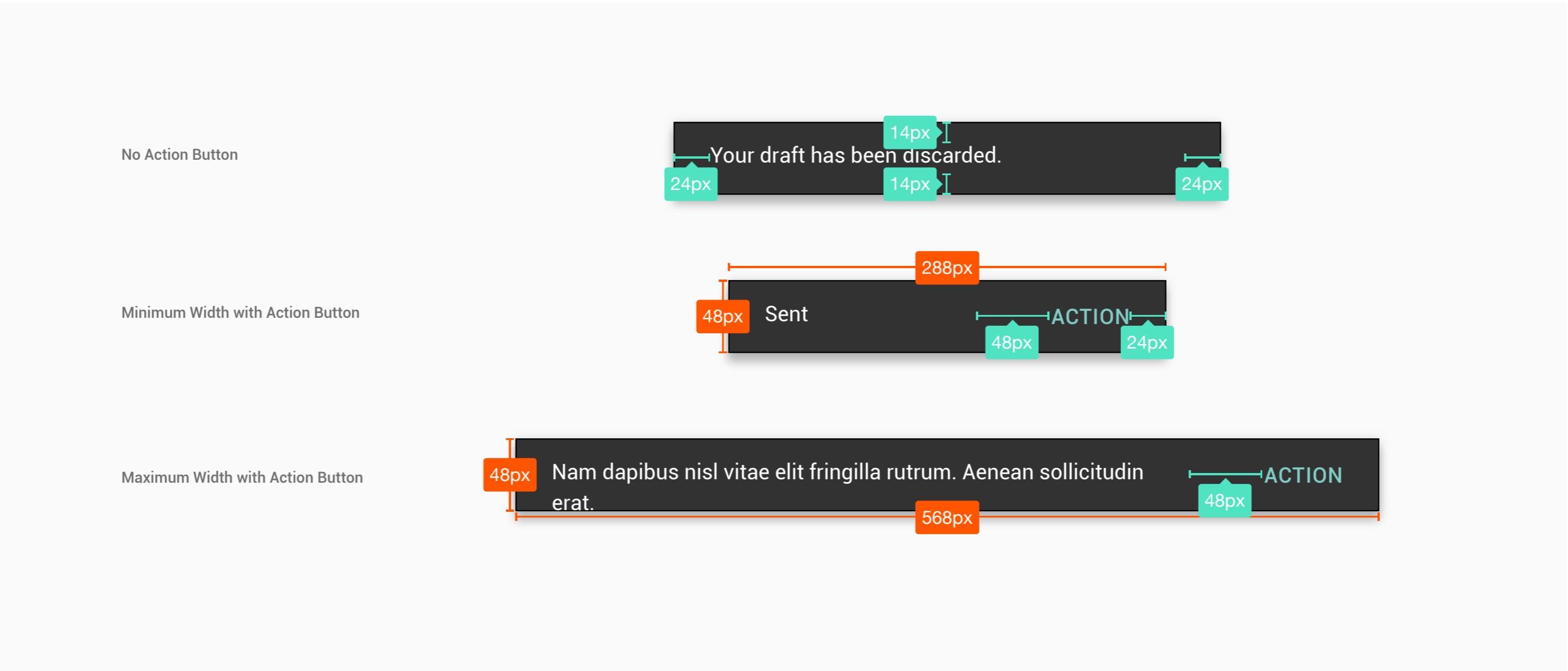
Default background fill: #323232 100%

Alignment: Centered or left-aligned 24px from the left and bottom edges of the screen

Text

Text: Roboto Regular 14px

Action button: Roboto Medium 14px, all-caps text



13.3 VARIATIONS

Here are some examples of exceptions and variations to the guidelines above.

13.4 REASONING

Why have we chosen to implement tool tips the way we’ve stated above.

TABS

Tabs make it easy to explore and switch between different views.

Tabs enable content organization at a high level, such as switching between views, data sets, or functional aspects of an app. Present tabs as a single row above their associated content. Tab labels should succinctly describe the content within.

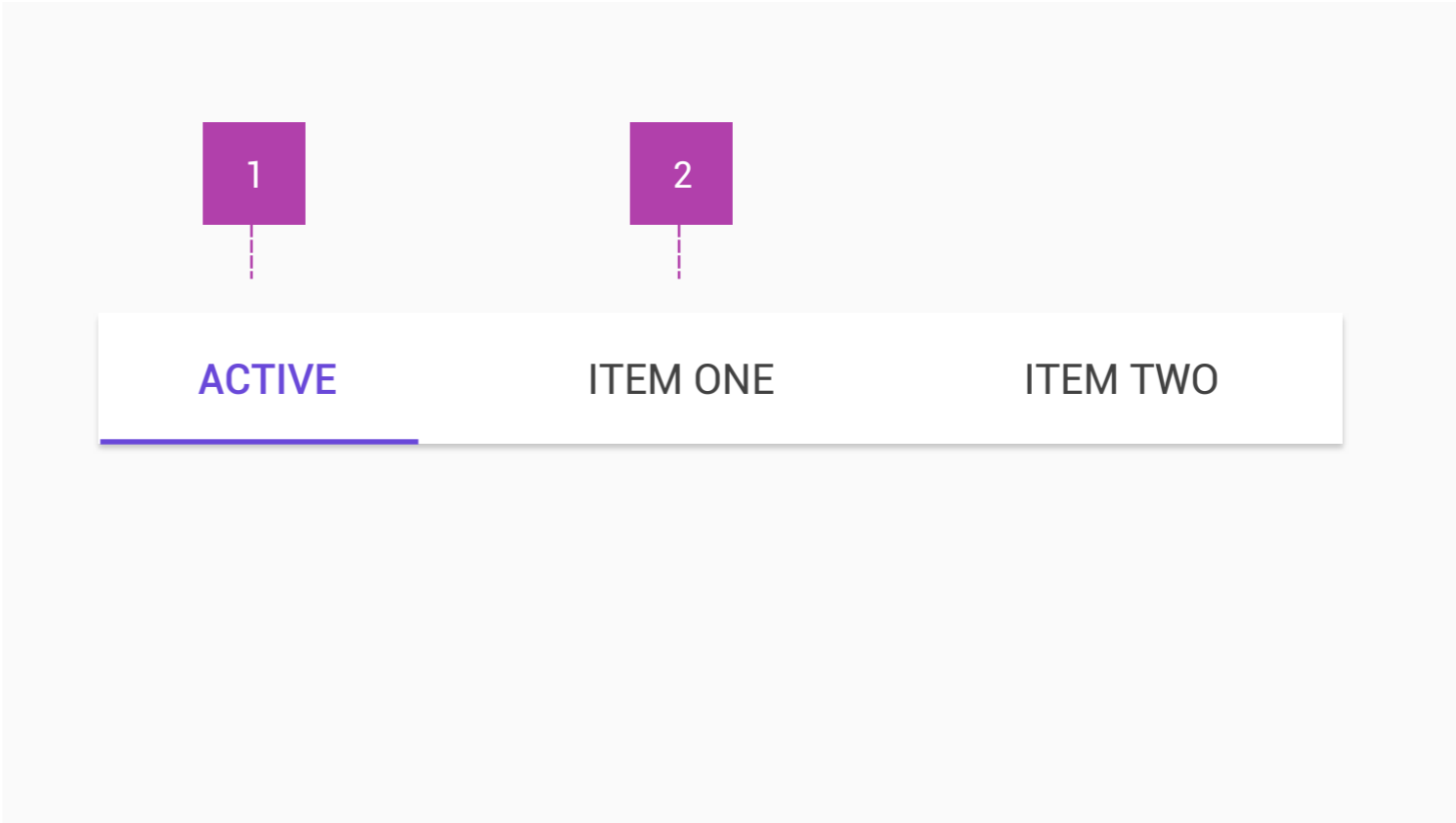
USAGE

A tab provides the affordance for displaying grouped content. A tab label succinctly describes the tab's associated grouping of content

STRUCTURE

Tabs are made up of the following components:

- 1. Active tab that indicates the active menu item the user is on.
- 2. Inactive tab that indicates the tab that the user has yet to take action with.



TYPES OF TABS

Depending on the platform and the context of use, tabbed content can be presented as either fixed tabs or scrollable tabs.

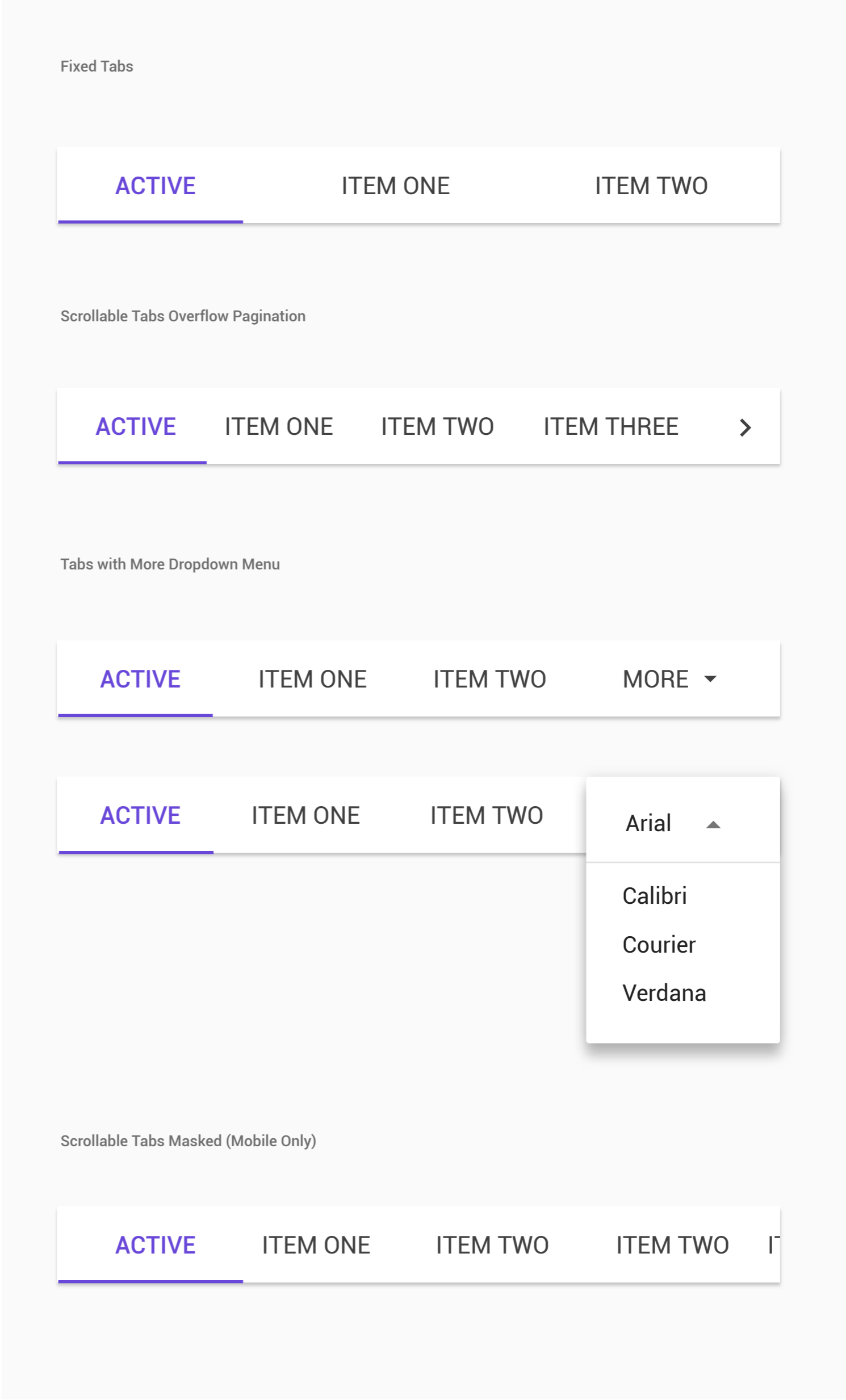
Fixed tabs

Fixed Tabs display all tabs concurrently and are best used with content that benefits from quick pivots between tabs in fixed positions, such as switching transportation methods for directions in Google Maps. Fixed tabs have equal width, calculated either as the view width divided by the number of tabs, or based on the widest tab label. To navigate between fixed tabs, touch the tab or swipe the content area left or right.

Scrollable tabs

Scrollable tabs display a subset of tabs at any given moment. They can contain longer tab labels and a larger number of tabs than fixed tabs. Scrollable tabs are best used for browsing contexts in touch interfaces when users don't need to directly compare the tab labels.

To navigate between scrollable tabs, touch the tab or swipe the content area left or right. To scroll the tabs without navigating, swipe the tabs left or right.



SPECIFICATIONS

Active Tab

Text: 16px Roboto Regular
Color: V100

Inactive Tab

Text: 16sp Roboto Medium
Color: C100

Active Tab line

Size: 2px border center
Color: C100

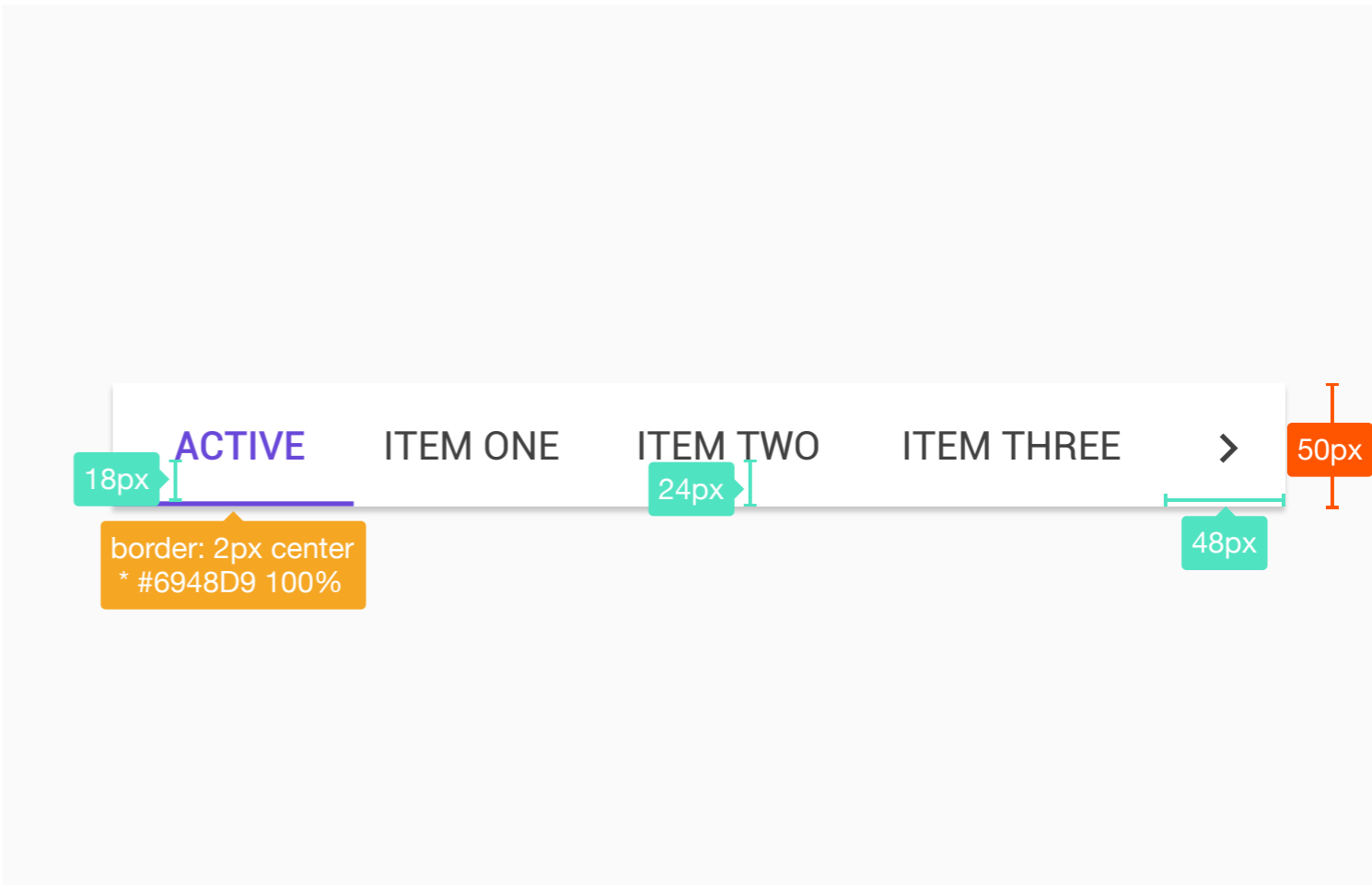
Inactive Tab line

Size: 2px border center
color: C100

Overflow / More Arrow Icons

Size: 24px x 24px
Color: C100

For more specs on dropdown refer to Dropdown component structure.



REASONING

Why have we chosen to implement fields the way we've stated above.

SEARCH

Search allows users to locate app content quickly.

When an app supports large amounts of information, users should be able to quickly locate content by searching for it.

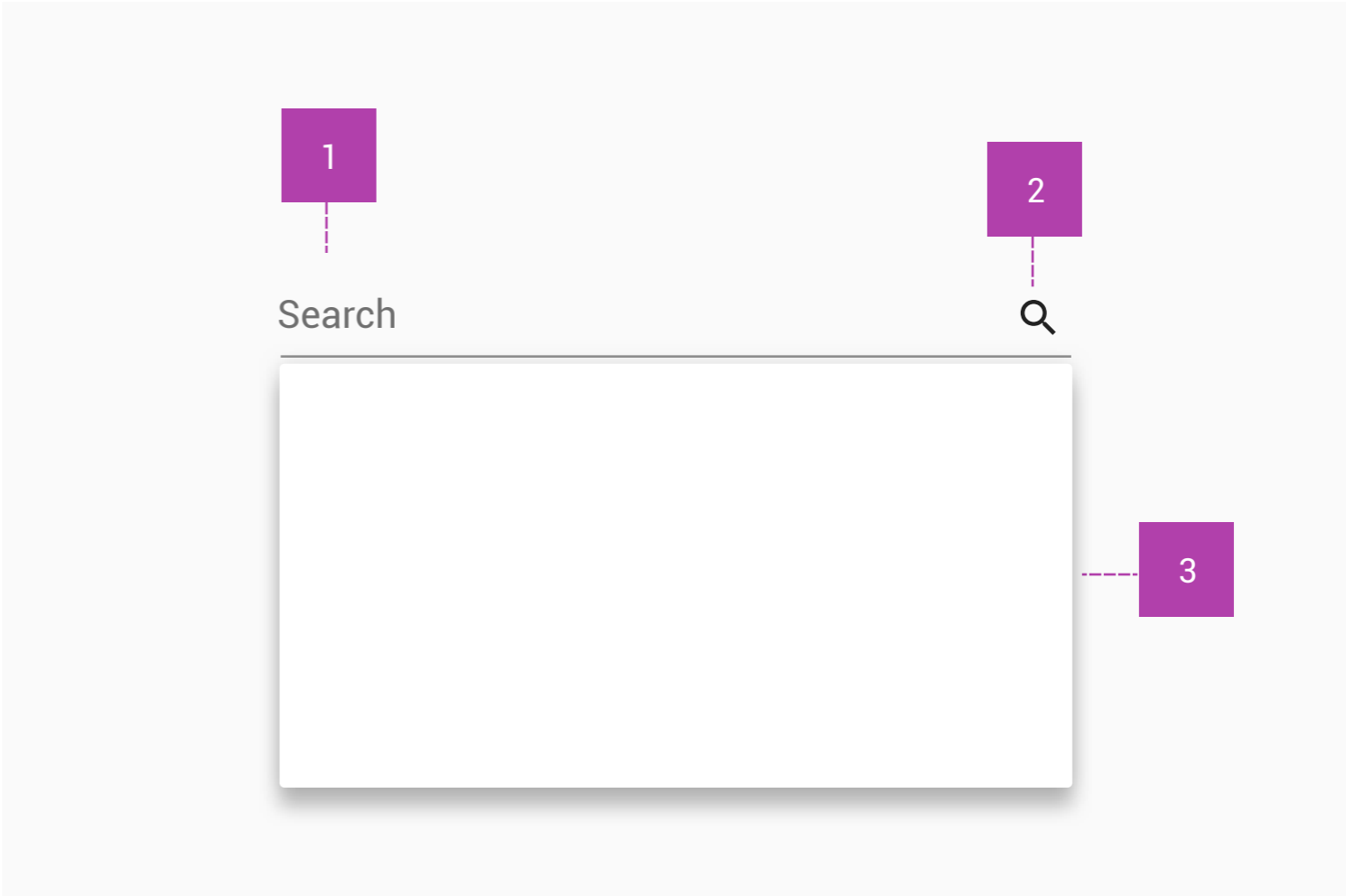
USAGE

Search can be used in either for the entire app (**global search**) or within a component (**component-based seacrh**). This variations allow more control as to the search results/autocomplete features that enable the user pin point data app wide or component wide.

STRUCTURE

Search is made up of the following components:

- 1. Search label/hint field area
- 2. Search icon that is also used as an action button to click for search
- 3. Search history, results and autocomplete area to display output



TYPES OF SEARCH

There are two types of search; Global and Component search. Both types can be used parallel to each other.

Global Search (also referred to as *persistent*)

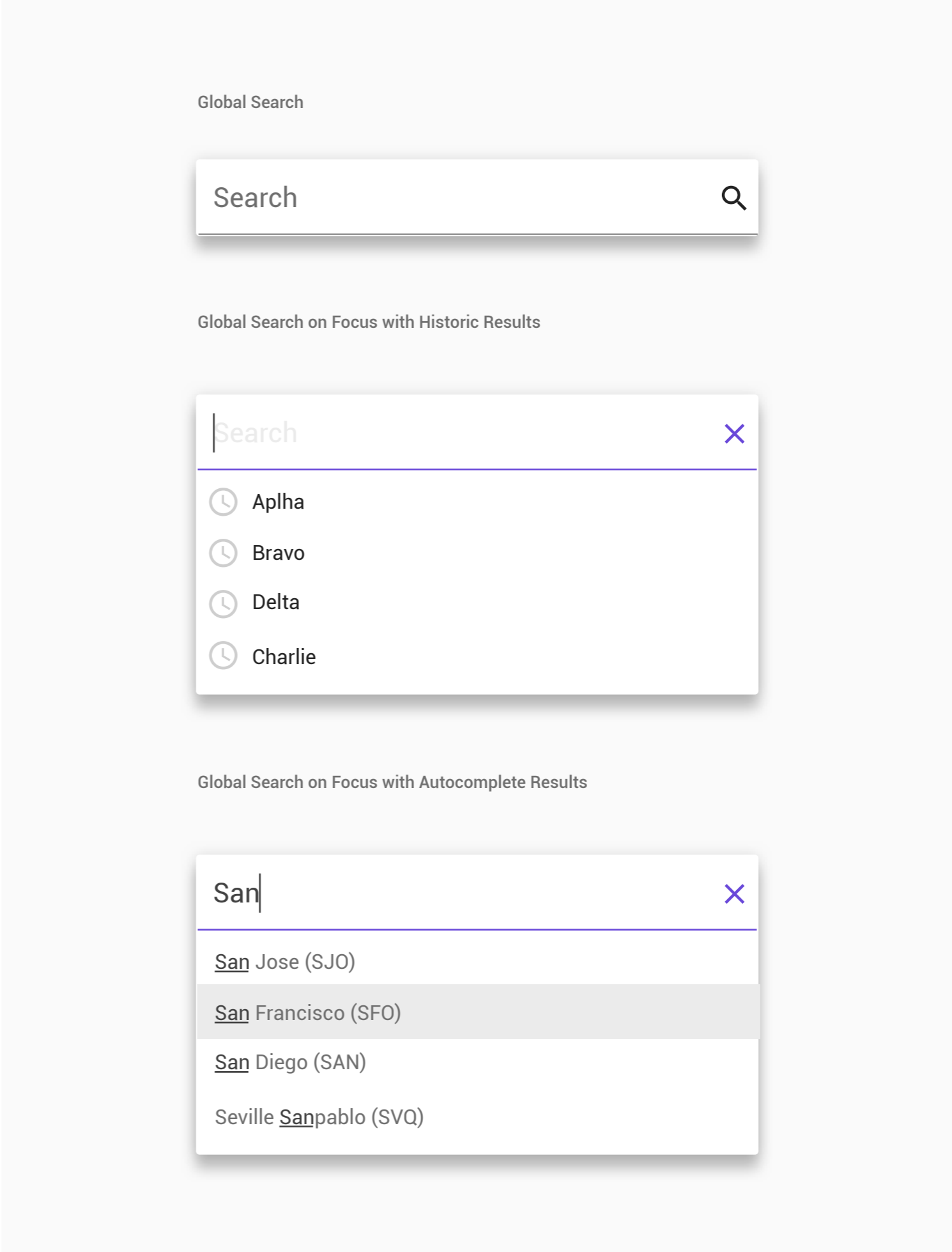
Enable global search when search is the primary focus and required within your app. Global Search should exist as part of the global navigation template

Behavior:

The search text field is presented inside the global navigation.

When in focus, the search field expands to show historical search suggestions and utilize autocomplete results.

Choosing any of the suggestions submits the search.



Component-based Search (also referred to as *expandable*)

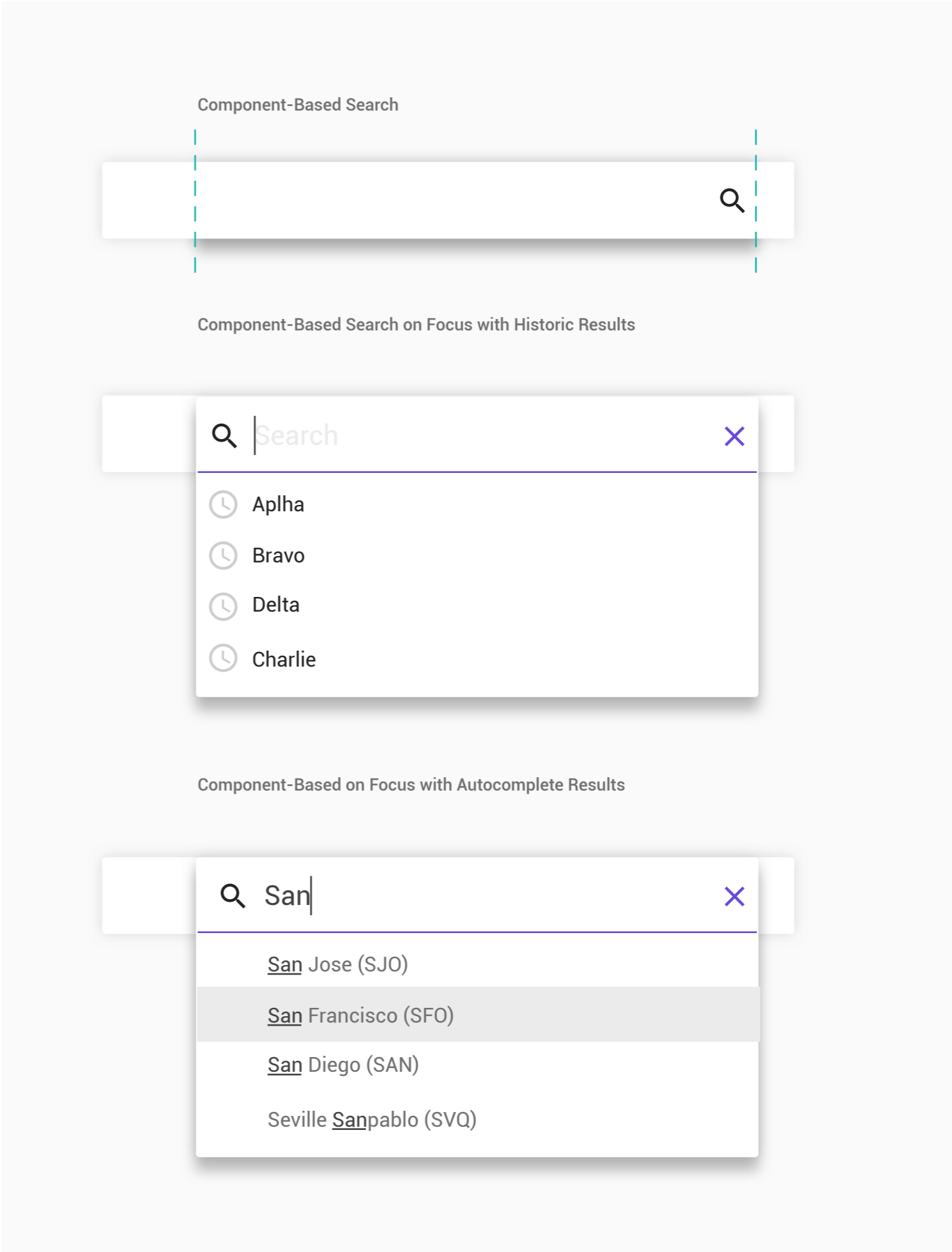
Use expandable search when you require to search within a specific component.

Behavior:

Display a magnifying glass icon in the toolbar of the card the component is constrained within.

Touching the search icon causes the toolbar to transform, clearing other content and displaying a search text field. If voice search is supported, a microphone icon also appears.

The search text field automatically receives focus. Historical search suggestions can be shown beneath the toolbar. Choosing any of the suggestions submits the query.



NOTIFICATIONS

Notifications are intended to inform users about events in your app. These two types of notifications are the most effective:

Communication from other users
Well-timed and informative task reminders
Notification components
Header area
Content area
Action area

USAGE

Notifications should not be the primary communication channel with your users, as frequent interruptions may cause irritation. The following cases do not warrant notification:

Cross-promotion, or advertising another product within a notification, which is strictly prohibited by the Play Store
An app that a user has never opened
Messages that encourage the user to return to an app, but provide no direct value, such as "Haven't seen you in a while"
Requests to rate an app
Operations that don't require user involvement, like syncing information
Error states from which the app may recover without user interaction

STRUCTURE

Overview

Notifications have been designed to make it easy to scan and use a notification's most important elements:

Primary content

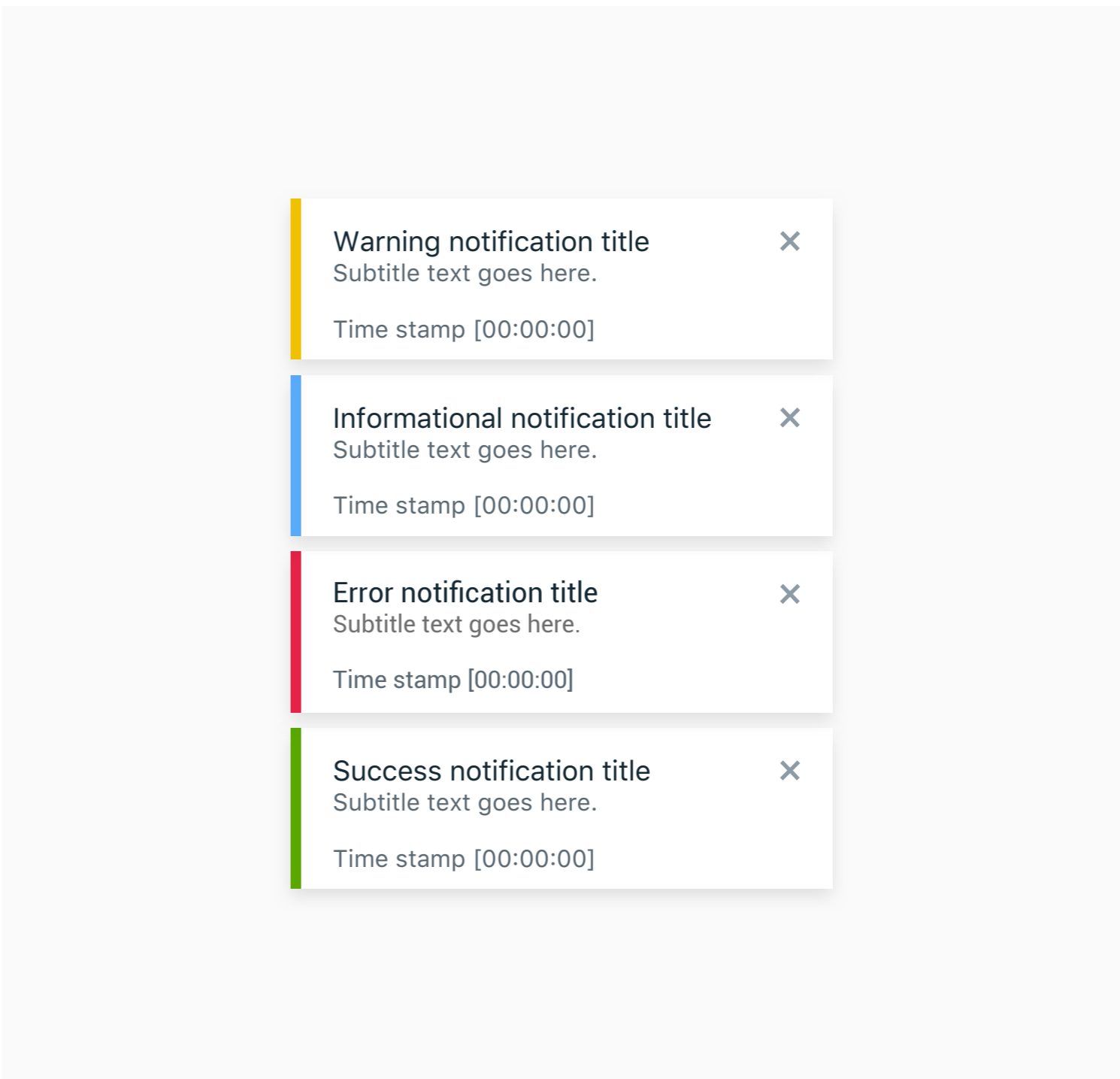
Content is the most prominent element of a notification. Secondary information, such as a timestamp, is smaller and consolidated above the primary content.

People

If the notification involves a person, an avatar appears on the right where it stands out from the rest of the content.

Actions

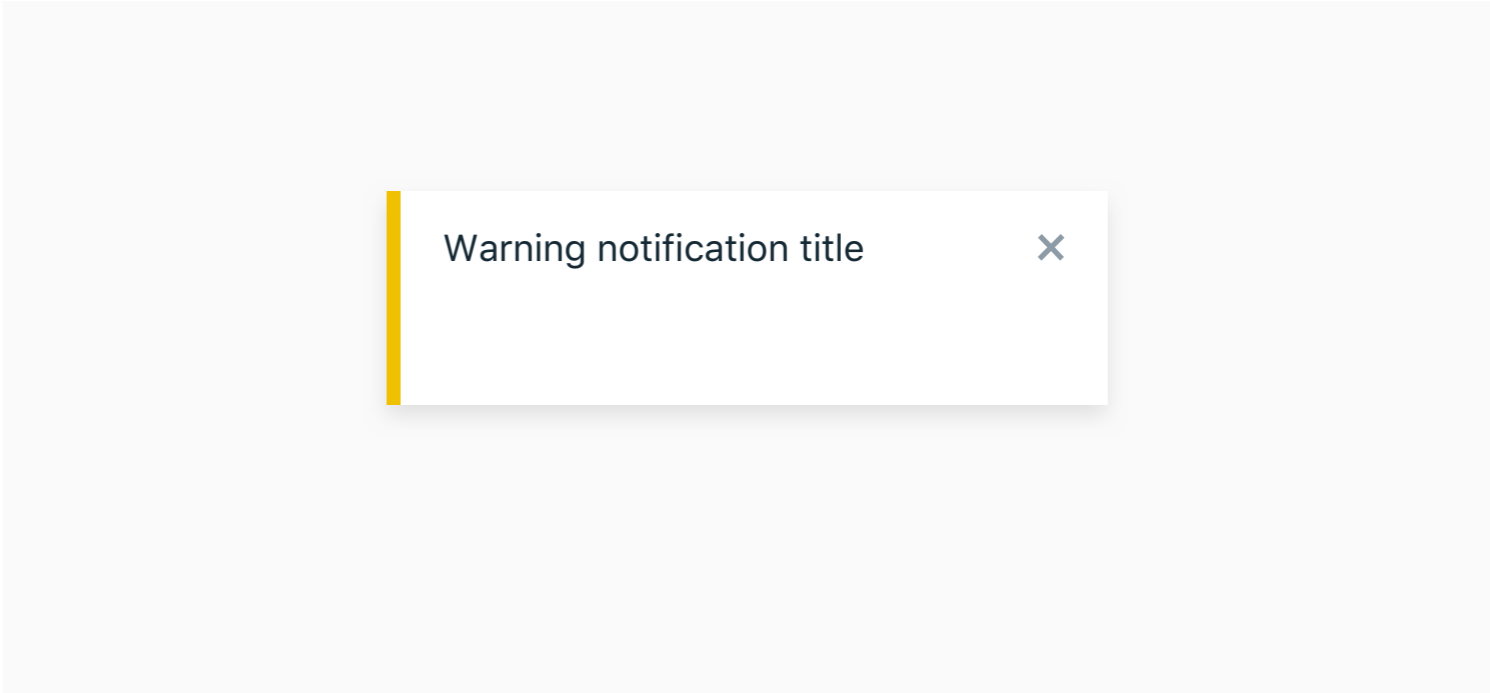
Expandable notifications are revealed by tapping an indicator icon. Actions are displayed with text labels on a separate background color and location.



Header area

The header area is comprised of:

1. The app icon: The app icon is a small two-dimensional representation of your app's identity. It appears in monochrome in the status bar. If your app sends a wide variety of notifications, you may replace your app's identity icon with a symbol that reflects the content type. For example, Google Now uses a cloud icon for weather notifications.
- 2.The app name: The name of the app automatically appears in the notification.
- 3.Header text (optional): Header text is usually only needed if an app sends notifications from multiple sources, such as the account name for users with multiple accounts.
4. Timestamp (optional): By default, a timestamp does not appear, but it may be added if it's important to display when a notification was sent, such as the time of a missed call.
5. Expand indicator: This appears if the notification can be expanded.



Content area

The content area is comprised of:

1. A content title: A brief headline for the notification
2. Content text: Supporting information
3. Large icon (optional): An image may be added to reinforce the notification in a meaningful way, such as a message that includes an avatar of the sender

Action area

When expanded, a notification may display up to three actions on at the bottom.

In Android N and later, actions are shown without icons to accommodate more text.

Required notifications

Foreground services are app processes that run in the background while the user is not directly interacting with your app. Because these apps use battery and possibly data, Android requires that users are made aware of these types of services by showing a non-dismissible notification.

Because the user can't dismiss the notification, you should provide an action for the user to stop the service if they don't want it running.

DIALOGUES

Dialogs inform users about a specific task and may contain critical information, require decisions, or involve multiple tasks.

Dialogs contain text and UI controls. They retain focus until dismissed or a required action has been taken. Use dialogs sparingly because they are interruptive.

Some dialog types include:

Alerts are urgent interruptions that inform about a situation and require acknowledgement.

Simple menus display options for list items, whereas simple dialogs can provide details or actions about a list item.

Confirmation dialogs require users to explicitly confirm a choice.

STRUCTURE

Beyond standard dialogs

Dialogs are a sub-type of modal windows, and the examples covered here are for standard material system dialogs. (Other modal window constructions aren't covered here because they have too much variation, such as branded buttons for purchasing flows, non-standard UI form elements, or unique layouts.)

Reduce interruption

Use dialogs sparingly because they are interruptive. Their sudden appearance forces users to stop their current task and focus on the dialog content. Not every choice, setting, or detail warrants interruption. Alternatives to dialogs include menus or inline expansion, both of which maintain the current context.

Dialog prominence

Dialogs should never be obscured by other elements or appear partially on screen. Dialogs always retain focus until dismissed or a required action has been taken, such as choosing a setting.

Dialogs should avoid:

- Opening dialogs from within a dialog
- Containing scrolling content
- Full-screen dialog exception
- Full-screen dialogs may open additional dialogs, such as pickers, because their design accommodates additional layers of material without significantly increasing the app's perceived z-depth or visual noise.

Scrollable content exception

Some dialog content requires scrolling, such as lists of ringtones.

For scrollable lists of options, the dialog title remains pinned to the top. This ensures that a selected item remains visible with the title, regardless of the item's position in the list.

Otherwise, the title scrolls off with the content. Actions always remain in place when content scrolls.

Dialogs are separate from the underlying parent material and do not scroll with it.

Displaying additional content

To disclose additional content in a dialog, do so using inline expansion within the content area. Or consider alternative components that are optimized for large amounts of content.

Dismissing dialogs

Dialogs may be dismissed either by tapping outside of the dialog, or tapping the system back button (on Android). Alternatively, the user's ability to dismiss a dialog may be disabled, so that one of the actions must be chosen before proceeding

Alerts

Alerts are urgent interruptions, requiring acknowledgement, that inform the user about a situation.

Disambiguation from Snackbars: Snackbars present optional information after an action, such as confirming the discarding of a draft. They often allow a user to undo an action just taken.

Alerts without title bars

Most alerts don't need titles.

They summarize a decision in a sentence or two by either:

- Asking a question (e.g. "Delete this conversation?")
- Making a statement related to the action buttons

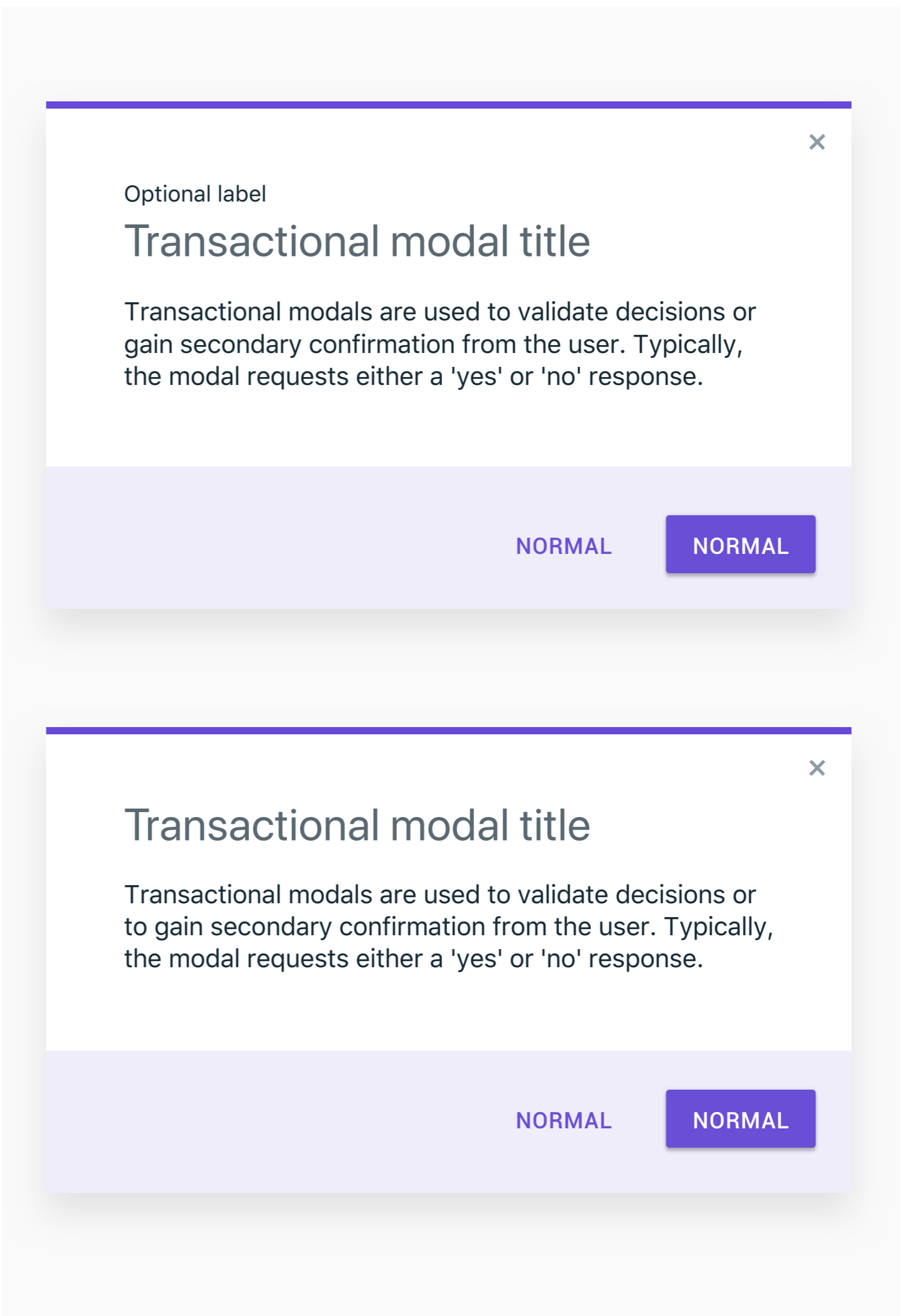
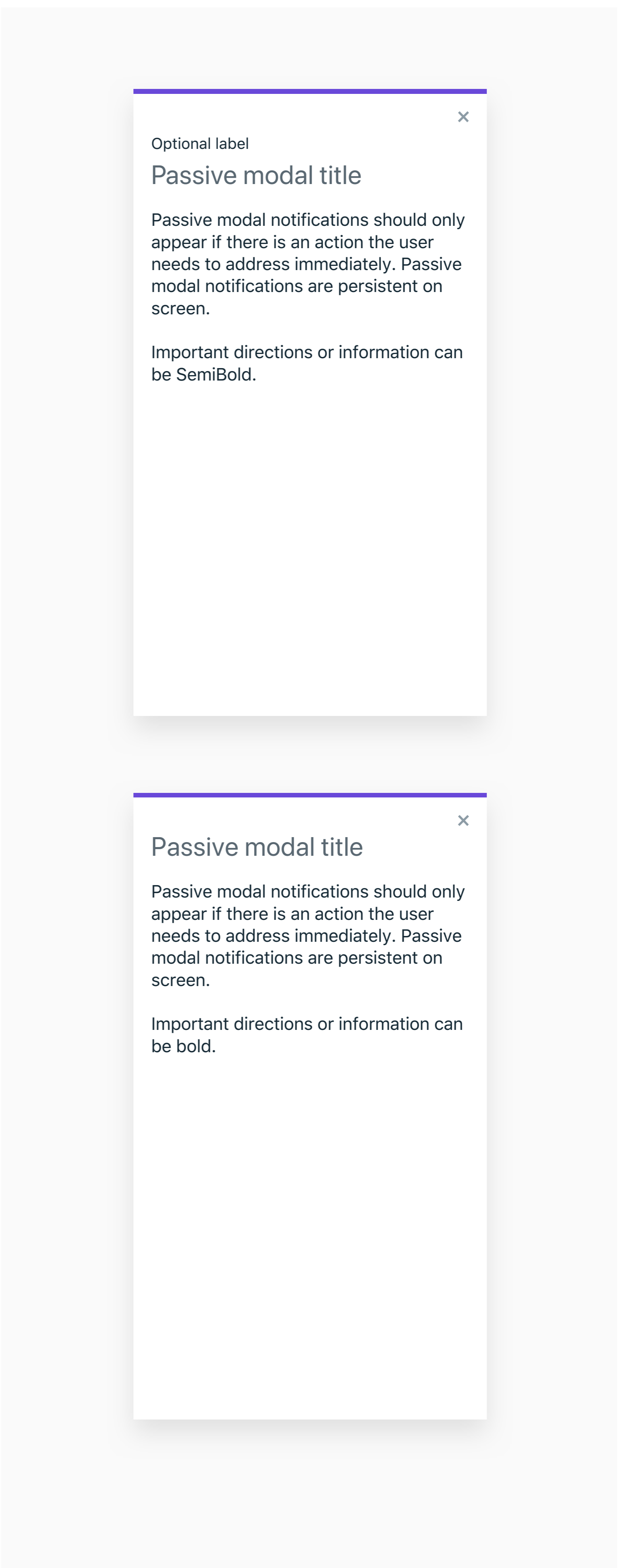
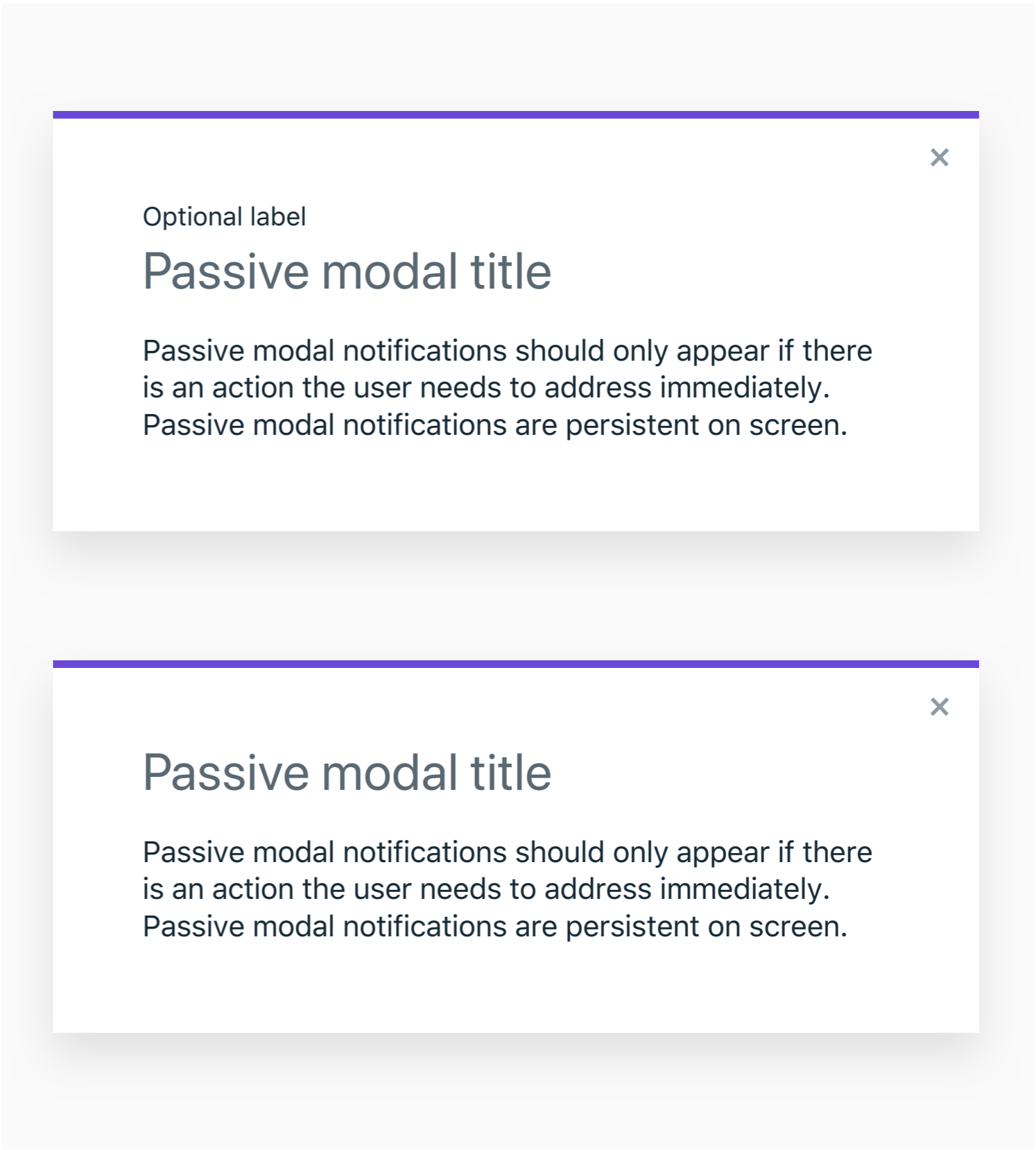
Alerts with title bars

Use title bar alerts only for high-risk situations, such as the potential loss of connectivity. Users should be able to understand the choices based on the title and button text alone.

If a title is required:

Use a clear question or statement with an explanation in the content area, such as "Erase USB storage?"

Avoid apologies, ambiguity, or questions, such as "Warning!" or "Are you sure?"



MOTION PRINCIPLES

 **STYLE**

Our visual signature for interfaces is based on cards which create the containers for any information displayed on the screen.

ICONOGRAPHY AND IMAGERY

 **STYLE**

Our visual signature for interfaces is based on cards which create the containers for any information displayed on the screen.

Our visual signature for interfaces is based on cards which create the containers for any information displayed on the screen.

Gutters

Screen width ≥ 768px = 20px gutters
Screen width < 768px = 10px gutters

Columns

Column count: 12

Margins

≥768px: 5% left, 5% right.
<768px: 3% left, 3% right.

Global Header Exception

The Global header lives outside the bounds of this grid. The outer margins of the global header is always 20px from the edge of the page.

Global header margin: 20px

Side Panel Exception

Any (left or right) side panel lives outside the bounds of this grid. The content area of the grid (including the grid margins) is indented width of the panel from the left edge of the page.

Grid area with left nav: page size minus 200px from left of page


Calculating Sketch layout settings for normal grid

Total Width:
width of artboard - [(width of artboard * margin %) * 2] = Total Width

Example:
Art board width = 1440px, margins @ 5%
1440 - [(1440 * .05) * 2] = 1296

Offset:
width of artboard * margin % = Offset

Example:
Art board width = 1440px, margins @ 5%
1440 * .05 = 72

 **Layout Settings**
These settings apply to selected Pages or Artboards only.

☒ Columns:

Total Width: 1296px

Offset: 72px

Center

Number of Columns: 12

☐ Gutter on outside

Gutter Width: 20px

Column Width: 90px

☐ Rows:

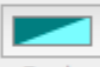
Gutter Height: 12px

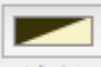
Row Height is 5 × Gutter Height

☐ Draw all horizontal lines

Visuals: ☐ Fill Grid☒ Stroke Outline

Colors:

 Dark

 Light

Make Default

Cancel

OK

Calculating Sketch layout settings for grid with left side panel

Total Width:


- width of artboard - width of left panel = content area
- (content area * margin %) * 2 = Y
- content area - Y = Total Width

Example:
Artboard width=1440px, Margins @ 5%; left nav width=200px

- 1440 - 200 = 1240
- (1240 * .05) * 2 = 124
- 1240 - 124 = 1116

Offset:
(content area * margin %) + width of left nav = Offset

Example:
Artboard width = 1440px; margins @ 5%; left nav width = 200px
(1240 * .05) + 200 = 262

 **Layout Settings**
These settings apply to selected Pages or Artboards only.

☒ Columns:

Total Width: 1116px

Offset: 262px

Center

Number of Columns: 12

☐ Gutter on outside

Gutter Width: 20px

Column Width: 75px

☐ Rows:


Gutter Height: 12px


Row Height is 5 × Gutter Height

☐ Draw all horizontal lines

Visuals: ☐ Fill Grid☒ Stroke Outline

Colors:

 Dark

 Light

Make Default

Cancel

OK

Calculating Sketch layout settings for grid with right side panel

Total Width:


- width of artboard - width of right panel = content area
- (content area * margin %) * 2 = Y
- content area - Y = Total Width

Example:
Artboard width=1440px, Margins @ 5%; right panel width = 288px

- 1440 - 288 = 1152
- (1152 * .05) * 2 = 115.2
- 1152 - 115.2 = 1036.8

Offset:
(content area * margin %) = Offset

Example:
Artboard width = 1440px; margins @ 5%
(1152 * .05) = 57.6

 **Layout Settings**
These settings apply to selected Pages or Artboards only.

☒ Columns:

Total Width: 1037px

Offset: 58px

Center

Number of Columns: 12

☐ Gutter on outside

Gutter Width: 20px

Column Width: 68px

☐ Rows:


Gutter Height: 12px

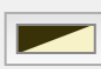
Row Height is 5 × Gutter Height

☐ Draw all horizontal lines

Visuals: ☐ Fill Grid☒ Stroke Outline

Colors:

 Dark

 Light

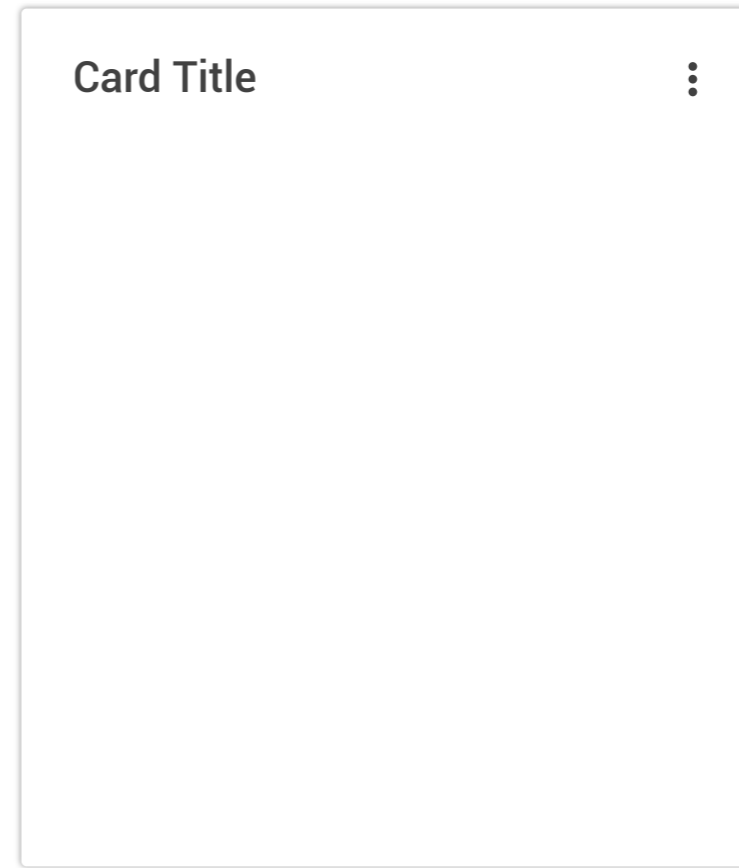
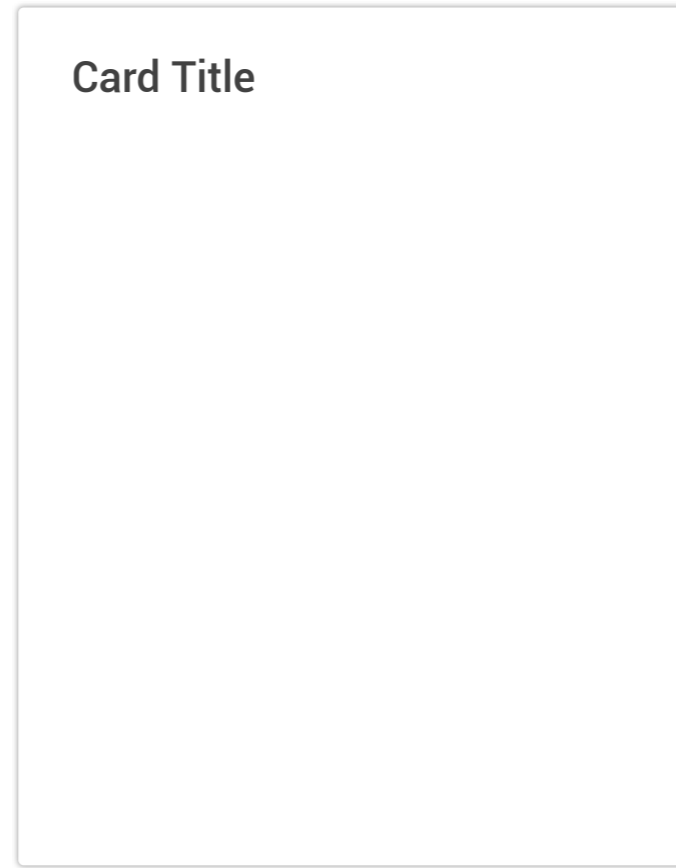
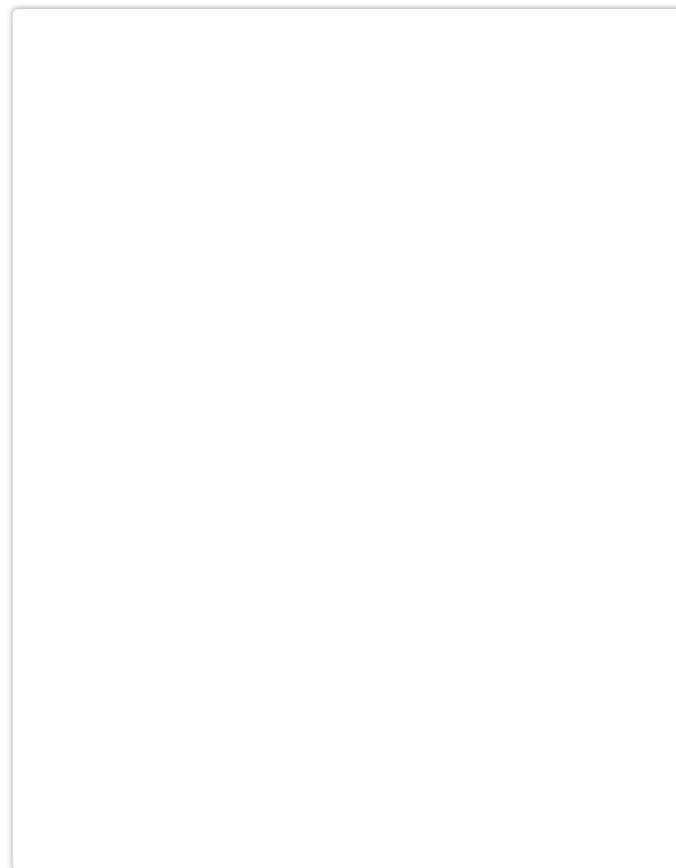
Make Default

Cancel

OK

LAYERS AND SPACING

Our visual signature for interfaces is based on cards which create the containers for any information displayed on the screen.



TEXT AREA

Text areas are taller than text fields and wrap overflow text onto a new line. They scroll vertically when the cursor reaches the bottom of the field.

Idle & empty	<div><div>Label</div><div></div></div> <div>Helper text</div>
Hover	<div><div>Label</div><div></div></div> <div>Helper text</div>
Focused	<div><div>Label</div><div>Placeholder</div></div> <div>Helper text</div>
Focus & Filled	<div><div>Label</div><div>Input</div></div> <div>Helper text</div>
Idle & Filled	<div><div>Label</div><div>Input</div></div> <div>Helper text</div>
Filled & Invalid	<div><div>Label</div><div>Input</div></div> <div>Error: Text explaining the error.</div>
Disabled	<div><div>Label</div><div>Input</div></div>

A dropdown button selects between multiple selections. The button displays the current state and a down arrow. Available states may be shown as a list of strings, a palette, or icons, for example.

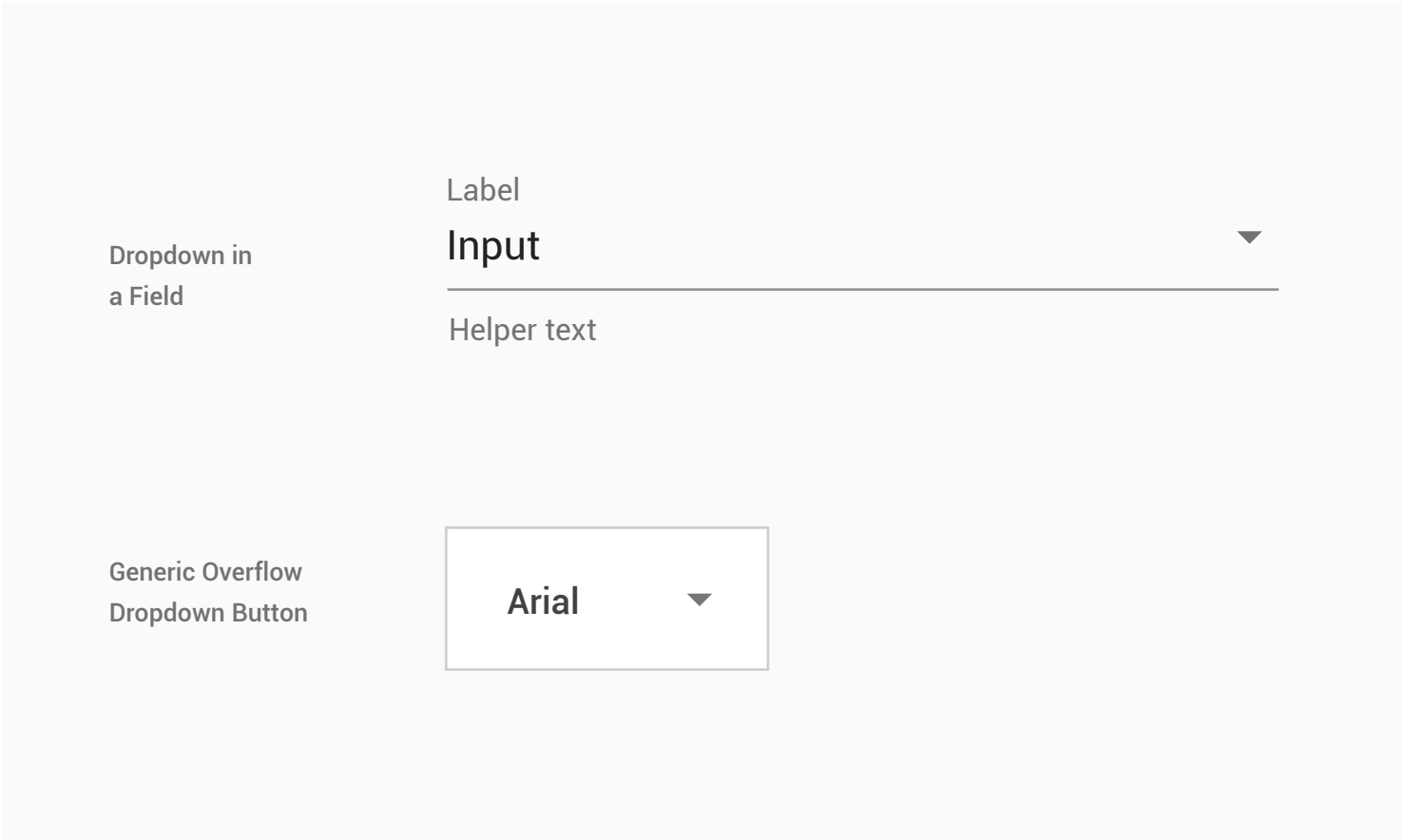
When a user interacts with the button, a menu covers the button and displays the possible states. Pressing a state dismisses the menu and updates the button to display this new state.

Scrolling within the dropdown behaves the same way a menu scrolls.

GENERIC OVERFLOW DROPDOWN BUTTON

The generic overflow dropdown button displays an arrow or menu button by default. When the button is pressed, the menu appears. Pressing an option on the menu navigates to further settings for that option.

This menu can appear in menus, in fields, or as stand-alone elements.



TIME PICKER

DEPRECATED

Time text:

- Type: Roboto Regular 60px
- Active Colour: W100
- Inactive color: W100, opacity: 70%

AM/PM text:

- Type: Roboto Medium 16px
- Active Colour: W100
- Inactive color: W100, opacity: 70%

Clock circle:

- Size: 296px x 296px
- Colour: C10

Clock text:

- Type: Roboto Regular 12px
- Colour: C100

Selected clock text:

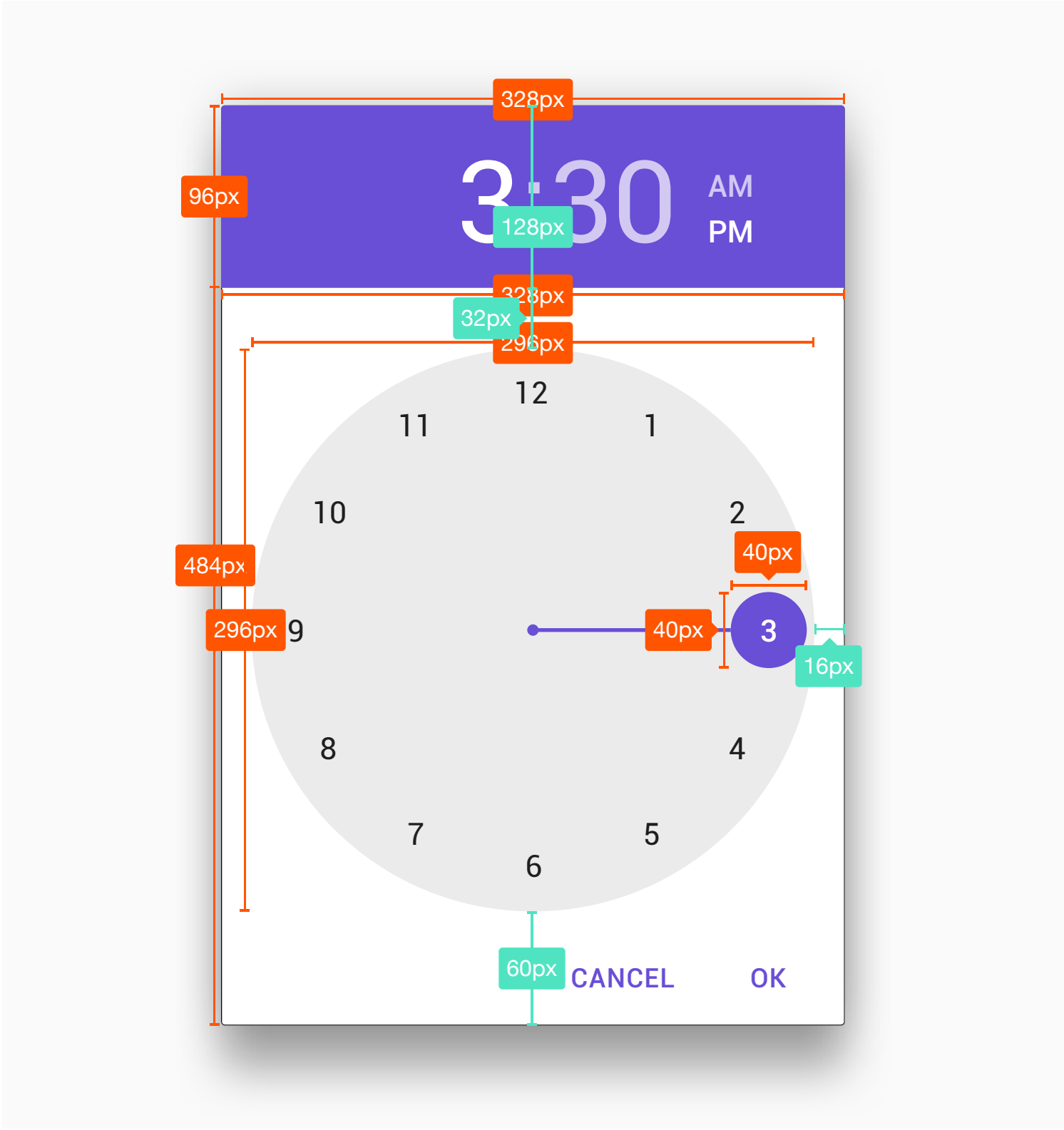
- Type: Roboto Medium 12px
- Colour: W100

Selected clock time circle:

- Size: 40px x 40px
- Colour: Primary color (V100)

Clock hand line:

- Size: 106px
- Stroke: 2px
- Color: Primary color (V100)



Label



Label

