



Oxyde UI

Development Release

This is a PDF version of the original AvdanOS toolkit linked in the AvdanOS Notion Site

The AvdanOS Team and Community

Sources

The links under are the following sources and affiliates for the entire document. AvdanOS is open source to the public and is free for all users. Please refer to the documentation for further information. AvdanOS is currently under development as of July 2022. We (the AvdanOS team) hold no liability nor does this project have any kind of legal warranty. In no event shall the authors (the AvdanOS team) liable for any claim or damages. This document was written 2022-07-12 and may not be up-to-date.

Links

AvdanOS Community Discord Server: <https://discord.gg/gRCcCUZ5px>

AvdanOS Website: <https://avdanos.com>

AvdanOS Documentation: <https://docs.avdanos.com>

AvdanOS GitHub: <https://github.com/Avdan-OS>

Avdan YouTube Channel: <https://www.youtube.com/c/Avdan>

AvdanOS YouTube channel: <https://www.youtube.com/channel/UCHLCBj83J7bR82HwjhCJusA>

For more social media, please refer to the AvdanOS website.

Configuring controls with variant properties

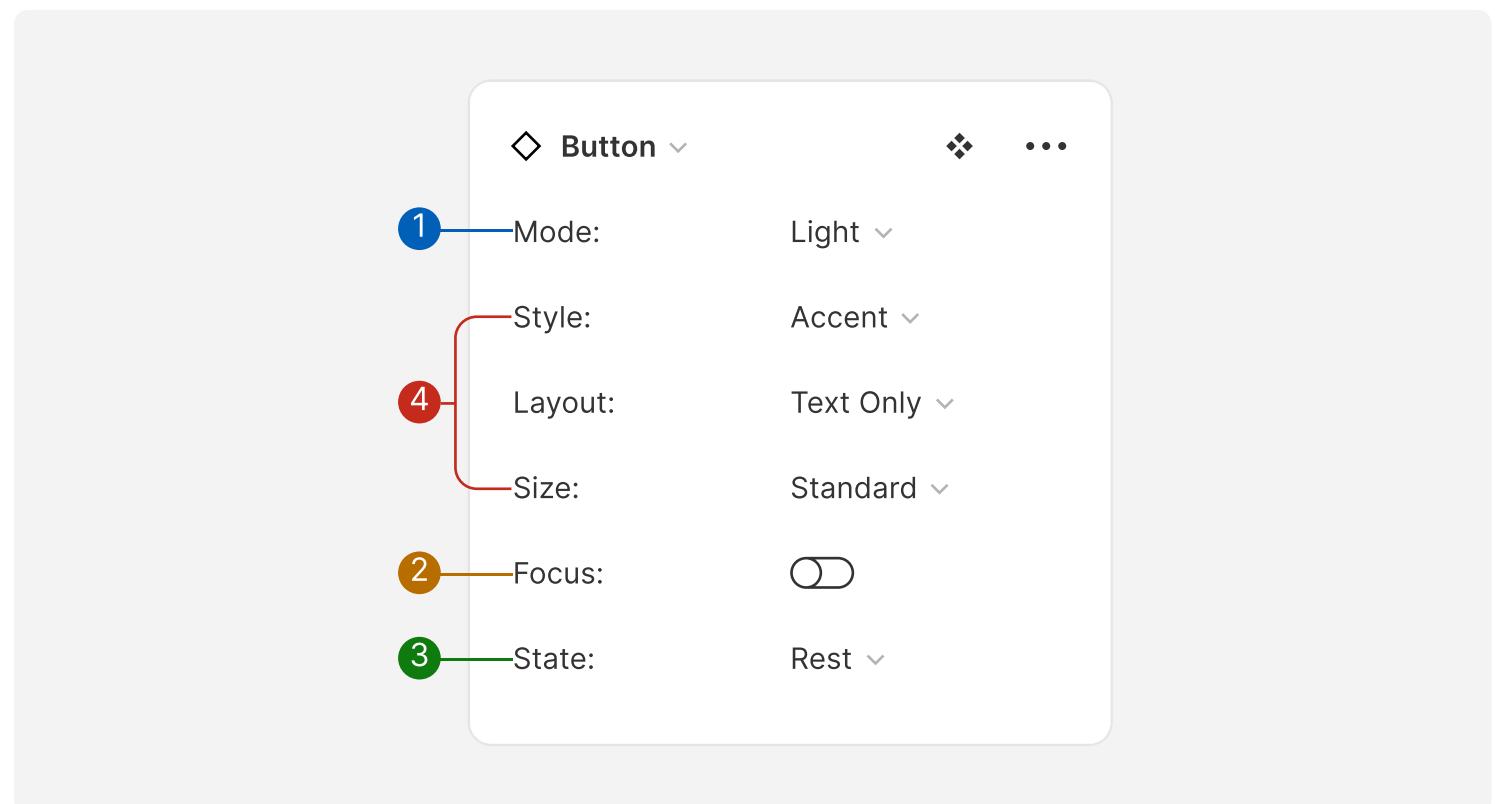
Each component has a set of common and control-specific variant properties that you can adjust to achieve a particular look and style of a component.

Common Properties

- 1 Mode: Light or Dark color modes 2 Focus: Toggles the keyboard focus visual (shown in the rest state only) 3 State: Rest, Hover, Pressed, Disabled, Selected Rest, Selected Hover, Selected Pressed, Selected Disabled

Specific Properties

- Properties that are specific to each component.



Instructions

This is the kit for creating AvdanOS experiences. It includes a component library plus guidelines and examples of how the components are used.

Getting the Controls

Assets Panel

Alternatively, you can always drag and drop a component into your design file from the Assets Panel in the left pane.

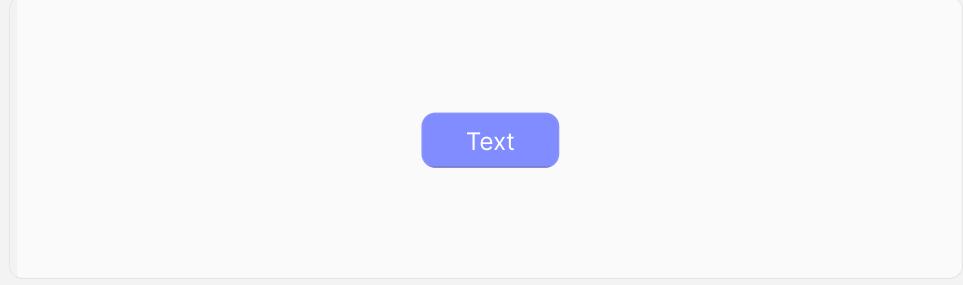
Copy me

On the “Controls” page, components are arranged with a “Copy me” section at the top of each component so that you can easily copy and paste it in your design.

Examples (copy and paste these too!)

Often it's much easier to start with a pre-assembled example and modify the example to suit your needs. Examples are provided for you to copy and paste them in your designs.

Copy me



Text

Understanding tags

Throughout the file informational tags are used to indicate control properties used, as well as any items that require customization of the control to achieve.

1 Variant Properties

These tags show which control specific variant properties are being used to render the look and style shown.

2 "Code Sample" Tag

This tag indicates when a particular variant is a customization of a control. While the rendered variant is not provided by the control by default, there is sample code available to replicate the look. Links to code samples can be found within the page title.

3 Custom Tag

This indicates elements that require customization of control code to achieve that do not currently have code samples available.

1

Variant Properties

Minimum Width

Fullscreen

Flyout Visible

2

Code Sample Tag

Code Sample

3

Custom Tag

Custom

What are primitives?

Primitives are Figma components that exist to represent common patterns. They are not WinUI controls. Typically, primitives are parts that are frequently used within control components, such as: material surfaces, icon buttons, and the components used for keyboard focus.

Standard screen sizes

Information pertaining to most commonly used screen sizes, snap points, DPI, scale factor, native and effective resolutions, and recommendations for designing app surfaces.

Basic Input

Button

A button gives the user a way to trigger an immediate action. Some buttons are specialized for particular tasks, such as navigation, repeated actions, or presenting menus.

Toggle Button

Represents a button that a user can select or clear.

Hyperlink Button

A button that's styled like a hyperlink and used for navigation.

Hyperlink

Provides an inline-level content element that provides facilities for hosting hyperlinks.

Dropdown

Represents a button with a chevron intended to open a menu.

Split Button

Represents a button with two parts that can be invoked separately. One part behaves like a standard button and the other part invokes a flyout.

Toggle Split Button

Represents a button with two parts that can be invoked separately. One part behaves like a toggle button and the other part invokes a flyout.

Combo Box

Use a combo box (also known as a drop-down list) to present a list of items that a user can select from. A combo box starts in a compact state and expands to show a list of selectable items.

Table of Contents

Guidance & Charts

Layering and Material

Windows uses a two layer system in order to create hierarchy and structure. When used consistently it will help users understand commanding and navigational elements while focusing on the content of your experience.

Elevation

Creating a visual hierarchy of elements in your UI makes the UI easy to scan and conveys what is important to focus on. Elevation, the act of bringing select elements of your UI forward, is often used to achieve hierarchy.

Typography

As the visual representation of language, typography's main task is to communicate information. Its style should never get in the way of that goal.

Windows Animation Values

Fluent motion serves a purpose in your app. It gives intelligent feedback based on the user's behavior, keeps the UI feeling alive, and guides the user's navigation through your app. Fluent motion elicits an emotional connection between a user and their digital experience. We build on a foundation of natural movement the user already understands from the physical world, and we extend our system from there.

Color

Color provides an intuitive way of communicating information to users in your app: it can be used to indicate interactivity, give feedback to user actions, and give your interface a sense of visual continuity.

Contrast

Contrast themes provide four customizable themes users can apply to increase visibility.

Toggle Switch

Use toggle switch to present users with two mutually exclusive options (such as on/off), where choosing an option provides immediate results.

Checkbox

Represents a control that a user can select (check) or deselect (uncheck). A Checkbox can also report its value as indeterminate (mixed state).

Radio Button

Radio button let users select one option from a collection of two or more mutually exclusive, but related, options.

Slider

A slider is a control that lets the user select from a range of values by moving a thumb control along a track.

Color Picker

A color picker is used to browse through and select colors.

Rating

The rating control allows users to view and set ratings that reflect degrees of satisfaction with content and services.

Lists & Collections

List View

Basic list item templates for use in a variety of controls.

Tree View

Represents a hierarchical list with expanding and collapsing nodes that contain nested items.

Flip View

Represents an items control that displays one item at a time, and enables "flip" behavior for traversing its collection of items.

Expander

An expandable and collapsible container to display content.

Grid View

Represents a control that displays data items in rows and columns.

Dialogs & Flyouts

Dialog

Dialog controls are modal UI overlays that provide contextual app information. They block interactions with the app window until being explicitly dismissed. They often request some kind of action from the user.

Flyout

A flyout is a light dismiss container that can show arbitrary UI as its content. Flyouts can contain other flyouts or context menus to create a nested experience.

Teaching Tip

A teaching tip is a semi-persistent and content-rich flyout that provides contextual information. It is often used for informing, reminding, and teaching users about important and new features that may enhance their experience.

Text Fields

Text Box

The TextBox control lets a user type text into an app. It's typically used to capture a single line of text, but can be configured to capture multiple lines of text. The text displays on the screen in a simple, uniform, plaintext format.

Password Box

Represents a control for entering passwords.

Rich Edit Box

You can use a RichEditBox control to enter and edit rich text documents that contain formatted text, hyperlinks, and images. You can make a RichEditBox read-only by setting its IsReadOnly property to true.

Auto Suggest Box

Represents a text control that makes suggestions to users as they enter text using a keyboard or pen (using ink and handwriting recognition). The app is notified when text has been changed by the user and is responsible for providing relevant suggestions for this control to display.

Number Box

Represents a control that can be used to display and edit numbers. This supports validation, increment stepping, and computing inline calculations of basic equations, such as multiplication, division, addition, and subtraction.

Scrolling

Scroll Bar (Scroll Viewer)

Scroll viewers enable content to extend beyond the bounds of the viewport (visible area). Users reach this content by manipulating the scroll viewer surface through touch, mousewheel, keyboard, or a gamepad, or by using the mouse or pen cursor to interact with the scroll viewer's scrollbar. This image shows several examples of scroll viewer controls.

Pips Pager

The Pips Pager control helps users navigate within linearly paginated content using a configurable collection of glyphs, each of which represents a single "page" within a limitless range. The glyphs highlight the current page, and indicate the availability of both preceding and succeeding pages. The control relies on current context and does not support explicit page numbering or a non-linear organization.

Status & Info

Badge

Badging is a non-intrusive and intuitive way to display notifications or bring focus to an area within an app - whether that be for notifications, indicating new content, or showing an alert. An InfoBadge is a small piece of UI that can be added into an app and customized to display a number, icon, or a simple dot.

Info Bar

The InfoBar control is for displaying app-wide status messages to users that are highly visible yet non-intrusive. There are built-in Severity levels to easily indicate the type of message shown as well as the option to include your own call to action or hyperlink button. Since the InfoBar is inline with other UI content the option is there for the control to always be visible or dismissed by the user.

Progress

A progress control provides feedback to the user that a long-running operation is underway. It can mean that the user cannot interact with the app when the progress indicator is visible, and can also indicate how long the wait time might be, depending on the indicator used.

Tooltip

A tooltip is a short description that is linked to another control or object. Tooltips help users understand unfamiliar objects that aren't described directly in the UI. They display automatically when the user moves focus to, presses and holds, or hovers the mouse pointer over a control. The tooltip disappears after a few seconds, or when the user moves the finger, pointer or keyboard/gamepad focus.

Menus & Toolbars

Context Menu (Menu Flyout)

Represents a flyout that displays a menu of commands.

File Menu (Menu Bar)

Menus and context menus display a list of commands or options when the user requests them. Use a menu flyout to show a single, inline menu. Use a menu bar to show a set of menus in a horizontal row, typically at the top of an app window. Each menu can have menu items and sub-menus.

Command Bar

Command bars provide users with easy access to your app's most common tasks. Command bars can provide access to app-level or page-specific commands and can be used with any navigation pattern.

Command Bar Flyout

The command bar flyout lets you provide users with easy access to common tasks by showing commands in a floating toolbar related to an element on your UI canvas.

Ink Toolbar

Represents a Universal Windows app control containing a customizable and extensible collection of buttons that activate ink-related features in an associated InkCanvas.

Navigation

Side Nav (Navigation View)

The Navigation View control provides top-level navigation for your app. It adapts to a variety of screen sizes and supports both top and left navigation styles.

Top Nav (Navigation View)

The Navigation View control provides top-level navigation for your app. It adapts to a variety of screen sizes and supports both top and left navigation styles.

Tab View

The Tab View control is a way to display a set of tabs and their respective content. TabViews are useful for displaying several pages (or documents) of content while giving a user the capability to rearrange, open, or close new tabs.

Breadcrumb

Breadcrumbs are a wayfinding element most appropriate when your data has a deep hierarchical structure with many levels.

Date & Time

Date Picker

The date picker gives you a standardized way to let users pick a localized date value using touch, mouse, or keyboard input.

Time Picker

The time picker gives you a standardized way to let users pick a time value using touch, mouse, or keyboard input.

Calendar Date Picker & Calendar View

A calendar view lets a user view and interact with a calendar that they can navigate by month, year, or decade. A user can select a single date or a range of dates. It doesn't have a picker surface and the calendar is always visible.

Media

Person Picture

The person picture control displays the avatar image for a person, if one is available; if not, it displays the person's initials or a generic glyph. You can use the control to display a Contact object, an object that manages a person's contact info, or you can manually provide contact information, such as a display name and profile picture.

Media Player

Media playback involves the viewing and listening of video and audio through inline (embedded in a page or with a group of other controls) or dedicated full-screen experiences.

Shell

Title Bar

When your app is running in a desktop window, you can customize the title bars to match the personality of your app. The title bar customization APIs let you specify colors for title bar elements, or extend your app content into the title bar area and take full control of it.

Primitives

List item

Reusable list item templates for use in a variety of controls.

Surfaces

Reusable background surfaces for use in app and shell contexts.

Caret

Reusable glyph for use in a variety of controls.

Focus rect

Reusable keyboard focus visual for use in controls.

Text active

Reusable active state for use with text controls.

Icon button

Reusable button component containing only an icon for its content.

Text box button

Reusable button component for use in a text box controls.

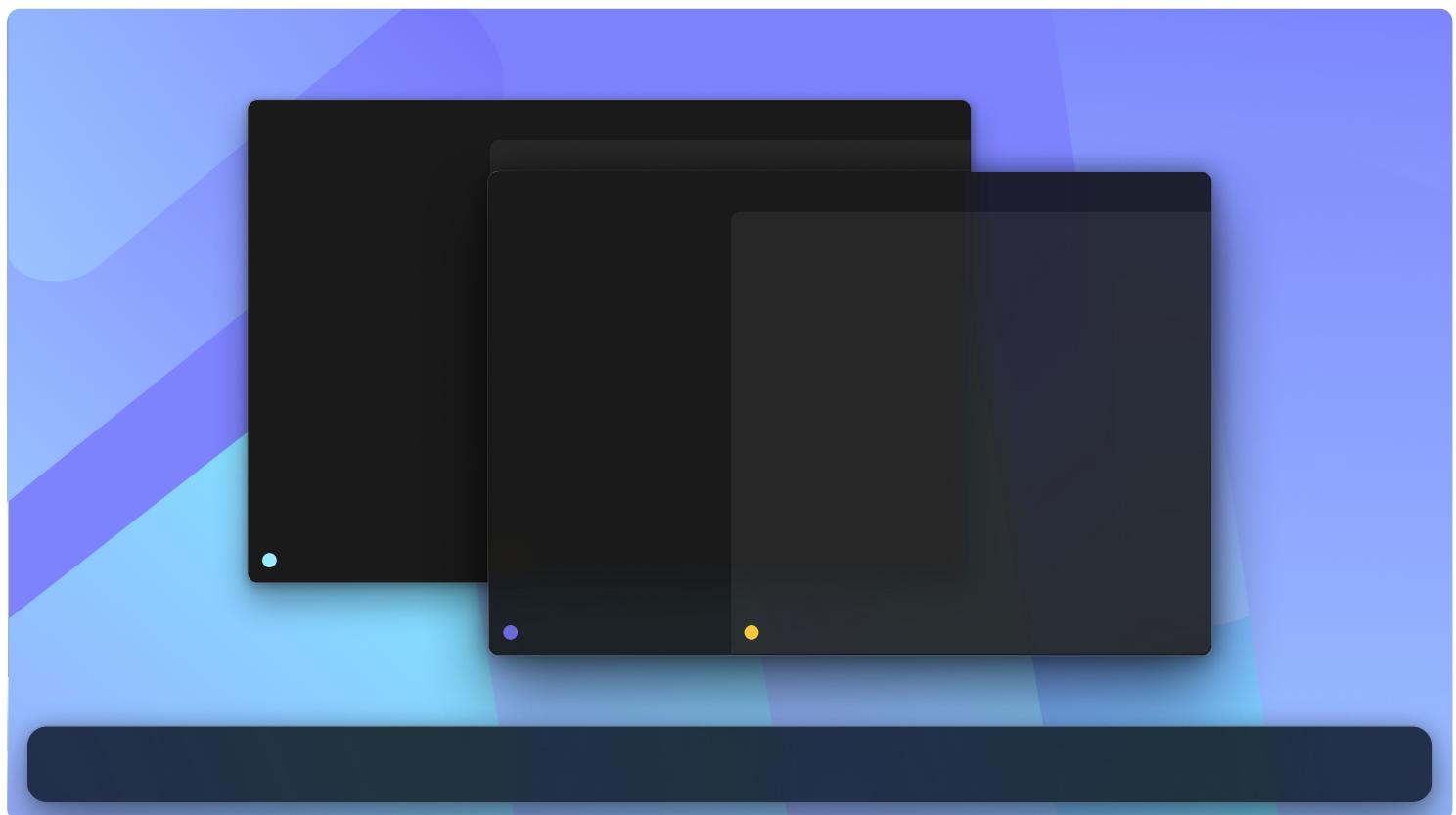
Date & time spinner button

The flipper or spinner button used by date and time picker flyouts to scroll columns.

App Surfaces

Biotite is used in applications and on app windows.

Biotite is a material that is translucent and blurred when active and opaque when inactive. Biotite integrates experiences into their surrounding environment, providing focus and guiding the user to the experience they are engaged with.



Component	Material	Usage
● App Surface / Base	Biotite (Base)	'Biotite Base' is the color style (used with Biotite Blur) used as the bottom layer for applications and windows.
● App Surface / Base (Inactive)	Solid (Base)	'Solid Base' is the color style (used with Biotite Blur) used as the bottom layer for applications and windows when they are not currently in focus.
● App Surface / Layer	Layer with opacity	'Layer Default' is the color style to be used when adding a second layer to 'Biotite Base'.

Layering and Material

Overview

AvdanOS uses a two layer system in order to create hierarchy and structure. When used consistently it will help users understand commanding and navigational elements while focusing on the content of your experience.

The back of your experience is the 'Base' — global or persistent commanding and navigation should be placed on this layer.

The front of panel is the 'Layer', this is where your content and page specific commanding lives.



- **Base**

Forms the foundation beneath your experience, containing global commands and navigation.

- **Layer**

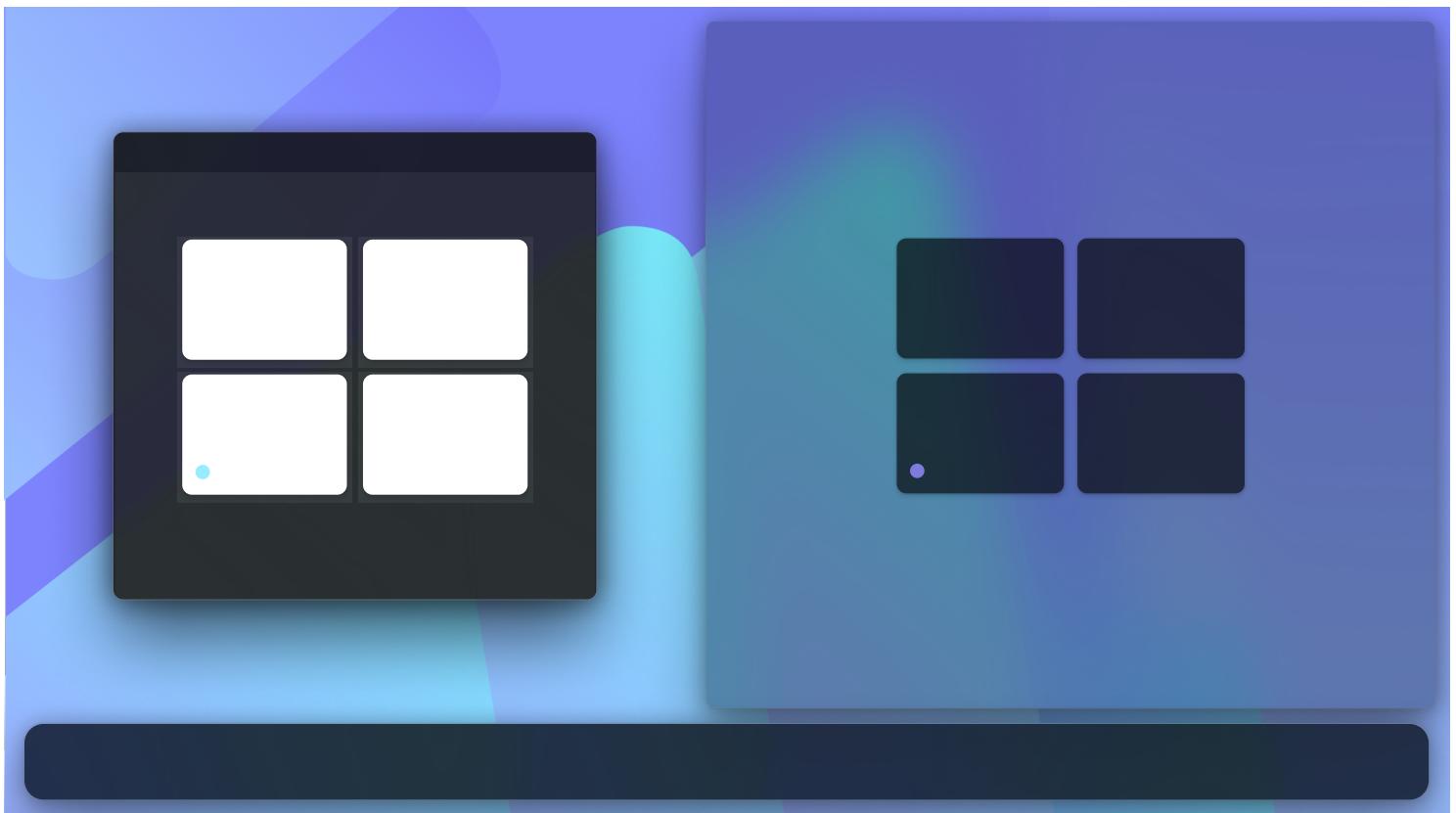
Serves to focus the user on the central experience. The content layer may be on contiguous elements, or separated into cards.

Card Surfaces

Cards are small layers that house content — they may be used on Base or Layer surfaces.

Standard cards come using 'card default' color -but you may also use 'Card Alt' if you need a secondary treatment.

There are also card recipes for use on 'Thin Acrylic' surfaces.



Component

Material

Usage

- **Card Surface (Standard)**

Layer with opacity

Optimized for being placed on 'Base', 'Layer', or 'Layer on Acrylic' backgrounds.

- **Card Surface (On Thin Acrylic)**

Layer with opacity

Optimized for being placed on Thin Acrylic surfaces.

Flyout Surfaces

A flyout is a lightweight contextual popup that displays UI related to what the user is doing. It includes placement and sizing logic, and can be used to reveal a secondary control or show more detail about an item.

Unlike a dialog, a flyout can be quickly dismissed by tapping or clicking somewhere outside the flyout, pressing the Escape key or Back button, resizing the app window, or changing the device's orientation.



Component

● Flyout Surface

Material

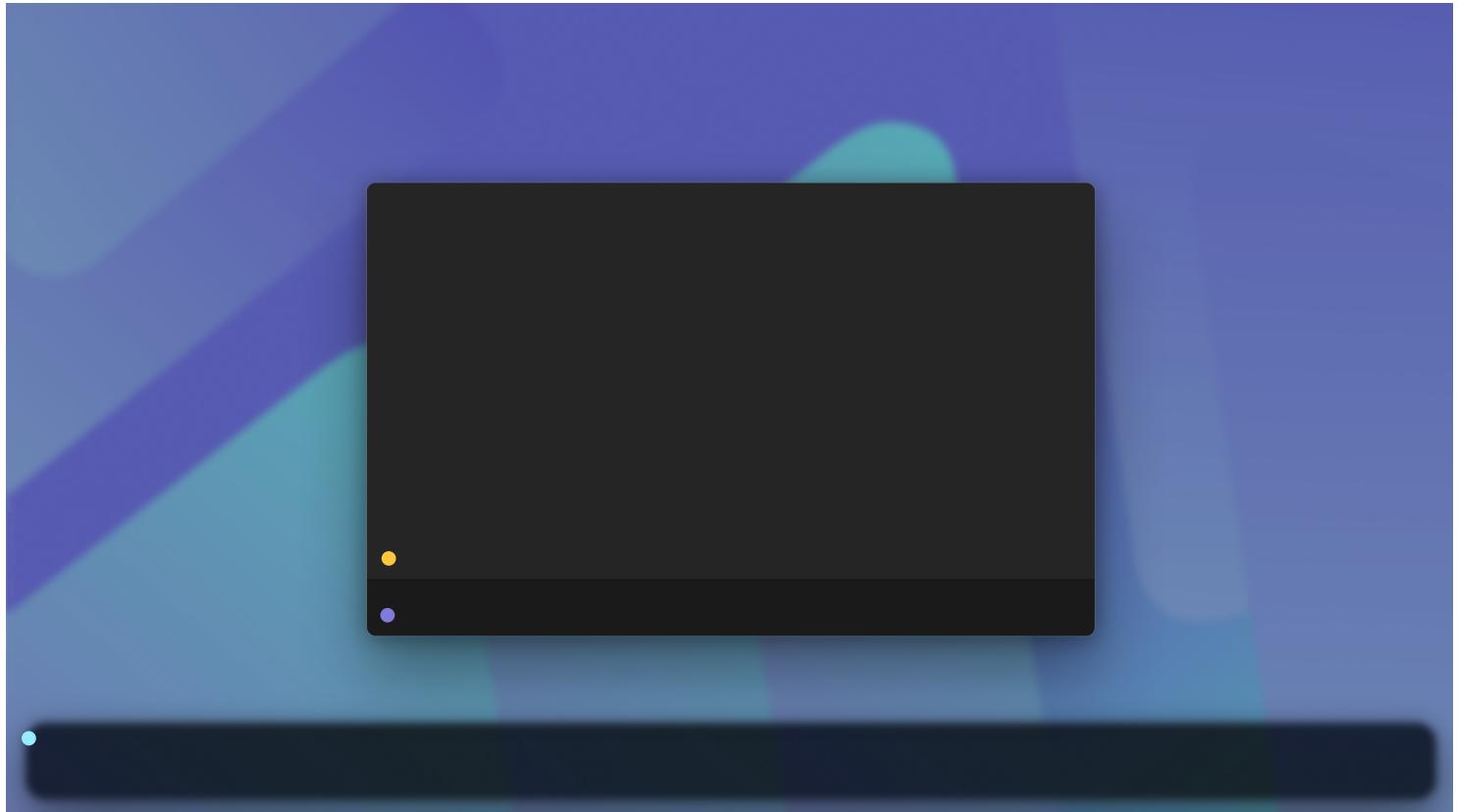
Acrylic (Default)

Usage

'Acrylic Default' is the recipe used for flyouts, menu overlays, and custom in-app surfaces experiences may wish to build.

Dialog Surfaces

Dialog surfaces are intended to focus the user on a specific moment or decision. The dialog itself uses a solid base and bright content layer paired with a darkening ‘smoke’ layer behind it. This makes the dialog ‘come forward’ visually and blocks out the content behind it.



Component

Material

Usage

● Dialog Surface / Smoke	Layer with opacity	Smoke is used for some modal experiences to provide focus on the task at hand by dimming content behind it. Smoke is to be used for ‘blocking’ dialogs that require action before dismissing.
● Dialog Surface / Base	Solid Layer	Dialogs, which appear on smoke, use ‘Solid Base’ as their fill color.
● Dialog Surface / Layer	Solid Layer	Dialog layers use ‘Layer Alt’ colors to make them stand out more from the dimmed background behind them.