

[AppKit](#) / View Management

API Collection

# View Management

Manage your user interface, including the size and position of views in a window.

## Topics

### Content Controllers

`class NSWindowController`

A controller that manages a window, usually a window stored in a nib file.

`class NSViewController`

A controller that manages a view, typically loaded from a nib file.

`class NSTitlebarAccessoryViewController`

An object that manages a custom view—known as an accessory view—in the title bar–toolbar area of a window.

### Split View Interface

`class NSSplitViewController`

An object that manages an array of adjacent child views, and has a split view object for managing dividers between those views.

`class NSSplitView`

A view that arranges two or more views in a linear stack running horizontally or vertically.

`class NSSplitViewItem`

An item in a split view controller.

## Stack View Interface

`class NSStackView`

A view that arranges an array of views horizontally or vertically and updates their placement and sizing when the window size changes.

## Tab View Interface

`class NSTabViewController`

A container view controller that manages a tab view interface, which organizes multiple pages of content but displays only one page at a time.

`class NSTabView`

A multipage interface that displays one page at a time.

`class NSTabViewItem`

An item in a tab view.

## Paged Interface

`class NSPageController`

An object that controls swipe navigation and animations between views or view content.

## Media Library Interface

`class NSMediaLibraryBrowserController`

An object that configures and displays a Media Library Browser panel.

## See Also

### User Interface

≡ Views and Controls

Present your content onscreen and handle user input and events.

≡ View Layout

Position and size views using a stack view or Auto Layout constraints.

☰ Appearance Customization

Add Dark Mode support to your app, and use appearance proxies to modify your UI.

☰ Animation

Animate your views and other content to create a more engaging experience for users.

☰ Windows, Panels, and Screens

Organize your view hierarchies and facilitate their display onscreen.

☰ Sound, Speech, and Haptics

Play sounds and haptic feedback, and incorporate speech recognition and synthesis into your interface.

📄 Supporting Continuity Camera in Your Mac App

Incorporate scanned documents and pictures from a user's iPhone, iPad, or iPod touch into your Mac app using Continuity Camera.