

# VPanel

## Overview

The VPanel `struct` is a SwiftUI view representing a vertical panel with advanced customization options. It includes features such as toolbar management, alignment, spacing, padding, conditional rendering based on the availability of a valid license, and the option to use a ScrollView.

## Example Code

```
public init(_ toolbarName: String = "",
            alignment: Alignment = .topLeading,
            horizontalAlignment: HorizontalAlignment = .leading,
            spacing: CGFloat? = 0,
            toolbarItems: Binding<[NavigationItem]> = Binding.constant([]),
            hiddenToolbarItems: Binding<Bool> = .constant(false),
            updateToolbar: Binding<Bool> = .constant(false),
            toolbars: ToolbarVisibility = [],
            padding: CGFloat = 5,
            useScrollView: Bool = false,
            @ViewBuilder content: @escaping () -> Content)
```

## Parameters:

- `toolbarName`: The name of the toolbar associated with this panel.
- `alignment`: The alignment of the content within the panel.
- `horizontalAlignment`: The horizontal alignment of the content within the panel.
- `spacing`: The spacing between items in the content.
- `toolbarItems`: A binding to an array of `NavigationItem` to populate the toolbar.
- `hiddenToolbarItems`: A binding that determines whether the toolbar items are hidden.
- `updateToolbar`: A binding that triggers a toolbar update when changed.
- `toolbars`: The visibility options for different toolbar elements.
- `padding`: The padding around the content within the panel.
- `useScrollView`: A flag indicating whether to use a ScrollView for the panel.
- `content`: A closure returning the content to be displayed in the panel.

## Example Usage

```
et verticalPanelWithScrollView = VPanel(
    "ExampleToolbar",
    alignment: .center,
    horizontalAlignment: .leading,
    spacing: 20,
    toolbarItems: Binding.constant([NavigationItem]()),
    hiddenToolbarItems: .constant(false),
    updateToolbar: .constant(false),
    toolbars: [.back, .forward],
    padding: 10,
    useScrollView: true
) {
    Text("Panel Content")
}

let contentView: some View = {
    VPanel(
        "ExampleToolbar",
        alignment: .center,
        horizontalAlignment: .leading,
        spacing: 20,
        toolbarItems: Binding.constant([NavigationItem]()),
        hiddenToolbarItems: .constant(false),
        updateToolbar: .constant(false),
        toolbars: [.back, .forward],
        padding: 10
    ) {
        Text("Panel Content")
    }
}()
```