ZPanel

Overview

The ZPanel 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 = .center,
                padding: CGFloat = 0,
                toolbarItems: Binding<[NavigationItem]> = Binding.constant([]),
                toolbars: ToolbarVisibility = [],
                updateToolbar: Binding<Bool> = .constant(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.
- padding: The padding around the content within the panel.
- toolbarItems: A binding to an array of `NavigationItem` to populate the toolbar.
- toolbars: The visibility options for different toolbar elements.
- updateToolbar: A binding that triggers a toolbar update when changed.
- content: A closure returning the content to be displayed in the panel.

Example Usage

```
let zPanel = ZPanel(
    toolbarName: "ExampleToolbar",
    alignment: .topLeading,
    padding: 10,
    toolbarItems: Binding.constant([NavigationItem]()),
    toolbars: [.back, .forward],
    updateToolbar: .constant(false)
) {
    Text("Panel Content")
}
```

```
let contentView: some View = {
    ZPanel(
        toolbarName: "ExampleToolbar",
        alignment: .topLeading,
        padding: 10,
        toolbarItems: Binding.constant([NavigationItem]()),
        toolbars: [.back, .forward],
        updateToolbar: .constant(false)
    ) {
        Text("Panel Content")
    }
}()
```