

[UIKit](#) / [View controllers](#) / Disabling the pull-down gesture for a sheet

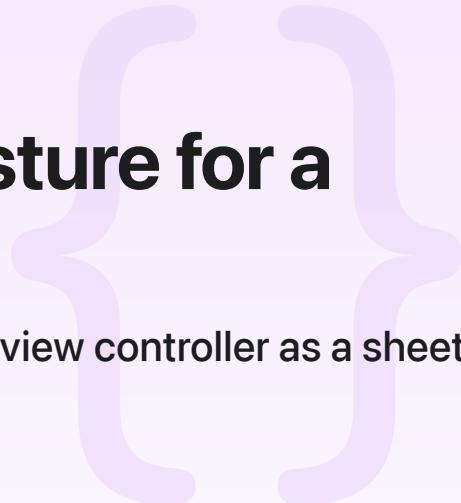
Sample Code

Disabling the pull-down gesture for a sheet

Ensure a positive user experience when presenting a view controller as a sheet.

[Download](#)

iOS 13.0+ | iPadOS 13.0+ | Xcode 12.3+



Overview

By default, a user can use a pull-down gesture to dismiss a view controller that presents as a sheet. UIKit allows you to disable the pull-down gesture in situations where using it might cause the user to lose data or recent changes. It's also possible to explain why the user is unable to dismiss the view controller presentation by presenting an instance of [UIAlertController](#).

Disable the ability to dismiss a presentation

To disable dismissal of a view controller presentation, set [`isModalInPresentation`](#) to true.

```
// If there are unsaved changes, enable the Save button and disable the ability to
// dismiss using the pull-down gesture.
saveButton.isEnabled = hasChanges
isModalInPresentation = hasChanges
```

It's also possible to return false from the presentation delegate's [`presentationControllerShouldDismiss\(_ :\)`](#) method. However, the system doesn't call this method if [`isModalInPresentation`](#) is true or when dismissing the presentation programmatically.

Explain why the user can't dismiss a presentation

To perform an action when the user attempts to dismiss a presentation that has a disabled dismissal, set the presentation's delegate as the code below shows:

```
// Set the editViewController to be the delegate of the presentationController for the presentation.
// editViewController can then respond to attempted dismissals.
navigationController.presentationController?.delegate = editViewController
```

After setting the delegate, implement `presentationControllerDidAttemptToDismiss(_)` and perform the action. The example below shows the presentation of an instance of `UIAlertController`:

```
func presentationControllerDidAttemptToDismiss(_ presentationController: UIPresentationController) {
    // The system calls this delegate method whenever the user attempts to pull down to dismiss and `isModalInPresentation` is false.
    // Clarify the user's intent by asking whether they want to cancel or save.
    confirmCancel(showingSave: true)
}

// MARK: - Cancellation Confirmation

func confirmCancel(showingSave: Bool) {
    // Present a UIAlertController as an action sheet to have the user confirm losing recent changes.
    let alert = UIAlertController(title: nil, message: nil, preferredStyle: .actionSheet)

    // Only ask if the user wants to save if they attempt to pull to dismiss, not if they tap the close button.
    if showingSave {
        alert.addAction(UIAlertAction(title: "Save", style: .default) { _ in
            self.delegate?.editViewControllerDidFinish(self)
        })
    }

    alert.addAction(UIAlertAction(title: "Discard Changes", style: .destructive) { _ in
        self.delegate?.editViewControllerDidCancel(self)
    })

    alert.addAction(UIAlertAction(title: "Cancel", style: .cancel, handler: nil))

    // If presenting the alert controller as a popover, point the popover at theCancelButton.
    alert.popoverPresentationController?.barButtonItem = cancelButton
}
```

```
present(alert, animated: true, completion: nil)
```

```
}
```

See Also

Presentation management

`class UIPresentationController`

An object that manages the transition animations and the presentation of view controllers onscreen.

`class UISheetPresentationController`

A presentation controller that manages the appearance and behavior of a sheet.