

Legacy - Troubleshooting the Alchemer Mobile iOS SDK

Here are a few of the more common issues that customers have run into when integrating the Alchemer Mobile iOS SDK in their app.

Help! My Love Dialog isn't showing!

This is probably the most common issue customers encounter. There are a few reasons why this could be happening:

- Not enough time has passed—or the app hasn't been launched enough times—since the current version of the app was installed: By default, the Love Dialog will not show in the first 5 days after installing a new app version, or before the app has been launched 5 times, as configured in the [Alchemer Mobile dashboard](#). Try turning off the Wait for setting, or set your device's clock forward the same number of days as you have specified. You can trigger a launch event in the simulator by selecting Home from the Hardware menu, and then clicking your app's icon to relaunch it. In any case, if you make a change on the dashboard, you will have to restart your app (and wait for "Received remote Interactions from Apptentive" to appear in the console) for the updated settings to take effect.
- *Who* or *Where* conditions are not being met: If you have stringent *Who* conditions for your Love Dialog, you may not be satisfying all of them when your *Where* event is engaged. Try reducing or removing all *Who* conditions until you see the love dialog, and then add the conditions back one at a time. Note that you will have to delete and reinstall the app for each configuration or the "only show once per version" logic will prevent you from seeing the Love Dialog a second time.
- The view controller passed in to the `-engage:fromViewController:` method isn't able to present UI: In version 3.2.0 and later of the SDK, you will see an error in the console when this happens ("Attempting to present interaction on a view controller whose view is not visible in a window."). This can happen for several reasons:
 - The `viewController` parameter is `nil`
 - The view controller has not been added to a window yet, for example if the `-engage:fromViewController:` method is called in `-viewDidLoad` (use `-viewDidAppear:` instead)
 - The view controller's view is no longer visible, for example if the `-engage:fromViewController:` method receives a view controller that is in the process of being dismissed or obscured. If you want to engage an event after dismissing a view controller, call `-engage:fromViewController:` in the completion block of `-dismissViewControllerAnimated:completion:` and pass in your view controller's `presentingViewController` property. Likewise, use the completion block of `-presentViewController:animated:completion:` to engage an event after presenting a view controller, passing in the newly-presented view controller.

The Apple Rating Dialog is not appearing or not working

There are several reasons why the Apple Rating Dialog might not show up during testing or in production:

- If you build the Alchemer Mobile SDK with a version of Xcode prior to 8.3, the Apple Rating Dialog can't be displayed. You can identify this in your logs by searching for the string `Apple Rating Dialog did not appear (reason: tools too old)`.
- If you are running on a device with an iOS version prior to 10.3, the Apple Rating Dialog is not supported. You can identify this in your logs by searching for the string `Apple Rating Dialog did not appear (reason: os too old)`.
- If you are running on a device where asking for ratings is disabled in the Settings app, or the user has been asked to rate three times already within the last year, or the user has already rated that version of the app, the Apple Rating Dialog will not appear. You can identify this in your logs by searching for the string `Apple Rating Dialog did not appear (reason: reached limit or user disabled)`.
- When testing using TestFlight, the Apple Rating Dialog is disabled. Instead, please test with a service such as [Microsoft App Center](#) or by running your app on a device attached to Xcode. In the latter case, the rating dialog will appear on every launch attempt, but the submit button will be disabled.
- When testing on a device running prerelease (Beta) software, reviews can't be submitted to the App Store.
- If the app was installed with a Promotional Code (i.e. if the user used the "Reedem" function in the App Store), a review can't be submitted.
- The Review Dialog (displayed if you tap "Write Review" after submitting a star rating from the Apple Rating Dialog) requires a nickname to be set for the account logged into the App Store in order for the "Send" button to work. Although the dialog gives you an opportunity to set a nickname, it appears to fail silently if you choose a nickname that is already in use. We recommend making sure you can successfully leave a review (for another app) in the App Store app before testing the Review Dialog.

Message Center isn't loading

In order to allow Message Center to be configured from the Alchemer Mobile dashboard, the SDK has to download the configuration from our API at least once before Message Center can be presented. This normally happens within 15 to 30 seconds of launching the app for the first time. By default, the SDK will log a message to the console indicating when the configuration has loaded ("Received remote Interactions from Apptentive.").

Before the configuration is loaded, users who launch Message Center will see a message explaining that it is not yet available ("We're attempting to connect. Thanks for your patience!"). In real-world use it is unlikely that a user would encounter this, as they would have to seek out the Message Center feature within a few seconds of launching your app for the first time. However it is

quite likely that you will encounter this when testing your app, so please wait for the configuration to load before launching Message Center.

Occasionally when our API is experiencing a period of unusually heavy load, it may take longer than the typical 15 to 30 seconds for the configuration to download. If you are seeing HTTP errors (specifically error number 502) in the console, or it is taking longer than a minute for the “Received remote Interactions from Apptentive” message to appear, please check our [Alchemer Mobile Status twitter account](#) for updates. In this case the Alchemer Mobile SDK will retry sending event, message, and survey data until it is successfully transmitted.

No attach button in Message Center

The image attachment button may be hidden for one of two reasons:

- Your app is not configured to support portrait mode on iPhone. This is required by the system image picker.
- Your app does not have an entry for the `NSPhotoLibraryUsageDescription` key (“Privacy – Photo Library Usage Description”) in its `Info.plist` file. iOS 10 and later requires a usage description for apps that wish to access the user’s photo library.

Unlike previous versions, when using the `UIImagePickerController`, you no longer need to explicitly request permission to access your customer’s files before they can send attachments.

My navigation bars are blank

By default, the `barTintColor` property of the navigation bars in the Alchemer Mobile UI is set to white. If your status bar text and icons, your navigation bar title text color, or your bar button items’ tint color is set to white as well, they will appear invisible. In that case you should make sure to set a contrasting bar tint color as described in the [customization guide](#).

Odd behavior when updating the SDK version

When updating the SDK from a prior version, occasionally the resource bundle can get out of sync with the SDK or vice versa. It may be helpful to remove files from Xcode’s Derived Data directory, as the Clean and Clean Build Folder options don’t always completely clear the appropriate caches:

```
rm -fr ~/Library/Developer/Xcode/DerivedData/*
```

Troubleshooting Push Notifications

The first and probably obvious thing to verify is that you are testing on an actual device, since push notifications are not supported in the Simulator.

Also, if an app is infrequently used, push notifications in the background may not appear or will only appear after a long delay.

Beyond that, we recommend testing the push capability of your app in isolation before moving on

and testing the Alchemer Mobile-specific aspects of receiving push notifications.

An excellent tool to help with the former is the [NWPusher](#) app, which you can build and run on your Mac to send push notifications directly to Apple's push servers. If everything is configured correctly, these notifications will appear on your device.

To use this tool you will need a push token. You can obtain this by setting a breakpoint in `application(_:didRegisterForRemoteNotificationsWithDeviceToken:)` and copying or logging the value of the `deviceToken` parameter. If this method is not called, make sure that you are calling the application object's `registerForRemoteNotifications()` method, and that the system is not calling the `application(_:didFailToRegisterForRemoteNotificationsWithError:)` method (if it is being called, the error can be instructive).

Next, verify that the certificate associated with your app is visible in NWPusher's dropdown menu, and that it is selected.

At this point you should be able to send a push to your app from NWPusher and have it received on the device. If this step fails, there is likely a problem or mismatch with the provisioning profile Xcode is using to build your app. It may also be the case that your app has not requested permission from the user to display banners, play sounds, or badge the app icon.

Once you have verified that your app can receive non-Alchemer Mobile push notifications, you will need to verify the following:

- The Remote Notifications background mode should be turned on for your app.
- The device token should be passed along to the Alchemer Mobile SDK in the `application(_:didRegisterForRemoteNotificationsWithDeviceToken:)` method.
- The remote notification should be passed along to the Alchemer Mobile SDK in the `application(_:didReceiveRemoteNotification:fetchCompletionHandler:)` method.
- The Alchemer Mobile SDK should be set as the current `UNUserNotificationCenter`'s delegate object, or if another object is already acting as the delegate, the `userNotificationCenter(_:didReceive:withCompletionHandler:)` and `userNotificationCenter(_:willPresent:withCompletionHandler:)` methods should be relaying any calls to the Alchemer Mobile SDK's `didReceiveUserNotificationResponse(_:from:withCompletionHandler:)` and `willPresent(_:withCompletionHandler:)` methods, respectively.
- The appropriate push certificates should have been exported from the Keychain Access app and uploaded to the Alchemer Mobile dashboard's Integrations section, and that push notifications are turned on there.

If none of these help with your push notification issue, please feel free to contact us.

Deep Links

Alchemer Mobile Prompts (formerly Notes) can have buttons that launch deep links, which can take customers to a specific page within your app. There are two important details to help with testing. These deep links typically follow a schema like: `myapp://path/to/content`

First, Alchemer Mobile does not currently support *universal* deep links. These typically look something like: <http://mydomain.com>

Second, if you're using a supported deep link and want to verify that it is a functional link, you can try opening that link on Safari mobile web. If the deep link opens the correct page of the app, then you know that it's working as intended. If it does not open the right page, then something about the deep link is incorrect. For example, if it opens the correct app but does not take you to the correct page of the app, then something about the deep link is wrong.

Related Articles