

Sample iOS SDK Migration Tasks

Sample Tasks for Migration

Below are two sample tasks to enter in your issue tracker for migrating the Alchemer Mobile iOS SDK.

Update the Alchemer Mobile Dependency to Version 6

Acceptance Criteria:

- Remove dependency for previous version(s) of the Alchemer Mobile iOS SDK
- Add dependency for the ApptentiveKit iOS SDK
- Update import statements from `import Apptentive` to `import ApptentiveKit`

Update from Deprecated Methods in the ApptentiveKit iOS SDK

For most customers, very few of these will need to be removed/updated. The simplest approach is likely to build your project and fix any deprecation warnings that result.

Acceptance Criteria:

Change references to newly-updated methods/properties

- Update from the `register(with:)` static method (that takes an `ApptentiveConfiguration` argument) to the `register(with:completion:)` instance method (that takes an `Apptentive.AppCredentials` argument)
- Update from `engage(_:fromViewController:)` to `engage(event:from:)`
- Update from `engage(_:fromViewController:completion:)` to `engage(event:from:completion:)` (note that the completion handler's argument's type is now `Result<Bool, Error>`)
- Update the completion handler in calls to `presentMessageCenter(from:completion:)` to accept a parameter of type `Result<Bool, Error>`
- (For Swift files) Update from `addCustomPersonData(_:withKey:)` and `removeCustomPersonData(withKey:)` to subscribing `personCustomData``
- Update code that accesses the `LogLevel` property to set the `LogLevel` on the `ApptentiveLogger` object or one of its static properties

Change references to newly-updated methods/properties

- Update from the `register(with:)` static method (that takes an `ApptentiveConfiguration`

argument) to the `register(with:completion:)` instance method (that takes an `Apptentive.AppCredentials` argument)

- Update from `engage(_:fromViewController:)` to `engage(event:from:)`
- Update from `engage(_:fromViewController:completion:)` to `engage(event:from:completion:)` (note that the completion handler's argument's type is now `Result<Bool, Error>`)
- Update the completion handler in calls to `presentMessageCenter(from:completion:)` to accept a parameter of type `Result<Bool, Error>`
- (For Swift files) Update from `addCustomPersonData(_:withKey:)` and `removeCustomPersonData(withKey:)` to subscribing `personCustomData``
- Update code that accesses the `LogLevel` property to set the `LogLevel` on the `ApptentiveLogger` object or one of its static properties

Change any remaining references to previously-updated properties

- Update references to the `mParticleId` property to `mParticleID`
- Update references to the `Apptentive.sharedConnection` static property to `Apptentive.shared`

Remove references to currently unavailable methods/properties

- Remove or comment out calls to `queryCanShowInteraction(forEvent:completion:)` (this feature is currently not supported)
- Remove or comment out calls to `queryCanShowMessageCenter(completion:)` (this feature is currently not supported)
- Remove or comment out calls to `dismissMessageCenter(animated:completion:)` (this feature is currently not supported)
- Remove or comment out calls to `openAppStore()` (this feature is currently not supported)
- Remove or comment out code that calls `login(withToken:completion:)`, `logout()`, or `updateToken(_:completion:)` or accesses the `authenticationFailureCallback` property (customer authentication is currently not supported)
- Remove or comment out calls to `setPushProvider(_:)`, `didReceiveRemoteNotification(_:fetchCompletionHandler:)`, `didReceiveLocalNotification(_:from:)`, `didReceiveUserNotificationResponse(_:from:withCompletionHandler:)`, and `willPresent(_:withCompletionHandler:)` (Push notification support will be restored in an upcoming release)
- Remove or comment out calls to `dismissAllInteractions(animated:)` (This feature will be restored in an upcoming release)

Remove references to removed methods/properties

- Remove the custom data argument from calls to `engage(_:withCustomData:fromViewController:)` (custom event data is currently not supported)
- Remove the extended and custom data arguments from calls to `engage(_:withCustomData:withExtendedData:fromViewController:)`, along with calls to methods that construct extended event data
- Remove code that accesses the `advertisingIdentifier` property (specific support for the advertising ID has been removed, but you can use custom person data instead)

- Remove code that accesses the `appId` property (App Store ratings now exclusively use the `SKStoreReviewController` class, which doesn't require the app ID)
- Remove code that accesses the `showInfoButton` property (the info button is no longer present)
- Remove code that accesses the `surveyTermsAndConditions` property (Survey terms and conditions are now configured in the dashboard)
- Remove code that reads the `apptentiveKey` or `apptentiveSignature` properties
- Remove code that accesses the `styleSheet` property
- Remove code that calls the `checkSDKConfiguration()` method

Update code that set properties on `ApptentiveConfiguration`

- `apptentiveKey` and `apptentiveSignature` are now passed when constructing the `Apptentive.AppCredentials` object passed to the `register(with:completion:)` instance method
- The `logLevel` property has moved to the `ApptentiveLogger` object
- Log messages are automatically sanitized when a debugger is not attached to the app
- The `baseUrl` can be set using the designated initializer for the `Apptentive` object (note that this prohibits using the `shared` instance of `Apptentive`)
- The `distributionName` and `distributionVersion` are now set on the `environment` property of the `Apptentive` object
- The `appId` property is no longer supported (see above)
- The `showInfoButton` property is no longer supported (see above)
- The `surveyTermsAndConditions` property is no longer supported (see above)

Related Articles