Legacy - Migrating Alchemer Mobile Android Versions

Certain versions of our SDK break compatibility with older versions. If you are migrating from an old version, please follow the migration guides below for each version between the version you are using and the version you are migrating to.

v5.0.0

Version 5.0 of our SDK has changed many of our synchronous API methods to be asynchronous. This will allow your app to perform better when using Alchemer Mobile (formerly Apptentive).

You will need to follow this if you were previously using a version of our SDK prior to 5.0, and you are:

- Using the return value of the Apptentive.engage() methods
- Using Apptentive.canShowInteraction()
- Using Apptentive push notifications

What changed?

In 5.0, we have made much of our API asynchronous to avoid slowing down your app. For most of our methods, you won't have to change anything in your app. But for some, where we return a value to you after performing a potentially time-consuming operation, we now return the value in a callback. And that requires an API change. Methods that have changed will now return void, and give you back a value through a callback when the operation has completed.

Please see our API Changes document for a complete list of method changes.

v4.1.0

Version 4.1.0 adds a troubleshooting mode that allows you to capture logs and send them to Alchemer Mobile's customer success team via email. In order to do this, we display our own "push" style notification that let's you send logs or exist this troubleshooting mode. In order to make this work if you are targeting API 26 with your app, we require you to use version 26 of the support libraries. Our SDK is compiled and tested with version **26.1.0** of the support library. You should always use the latest patch version of the support libraries that are available when you ship your app.

There are no API changes in this release.

v4.0.0

Please see the latest API Changes

Version 4.0.0 of the Alchemer Mobile Android SDK has new APIs for customer login, as well as some changed APIs for SDK registration. If you were using any previous version of the Alchemer Mobile SDK in your app, you will need to follow one or more of the applicable steps below.

Major Changes

Registration Has Changed

Registration now requires two pieces of information instead of one. Where before you needed to copy your API Key from the Alchemer Mobile (formerly Apptentive) dashboard, now you need to copy the Alchemer Mobile App Key and Alchemer Mobile App Signature. You can find both of those in the API & Development page.

Push Notification Changes

If you are not using the new Customer Login feature, you don’t strictly have to follow this step, but it is a good idea in case you add it in the future.

When you reply to your customers from the Alchemer Mobile website, we will send a push, if configured. If the customer for which the push is intended is not logged in, then the push should not be displayed. This small change means that you need to check whether the push is and Alchemer Mobile push, or should be handled by your existing push code, and then check where it can be displayed. See the Push Notifications documentation for code that demonstrates this change for the push provider you are using.

How to migrate

The migration process should take 5-10 minutes to complete.

1. Replace the call to `Apptentive.register(Application yourApp, String apiKey)` with a call to `Apptentive.register(Application yourApp, String apptentiveKey, String apptentiveSignature)`.
2. If you were calling `Apptentive.register(Application yourApp)`, and specifying the API Key in your manifest, you can continue to do so in version 4.0.0, you will just need to remove the old manifest element and add two new elements for the Alchemer Mobile (Apptentive) App Key and Alchemer Mobile (Apptentive) App Signature.
   Remove:
   `<meta-data android:name="apptentive_api_key" android:value="<YOUR_API_KEY>">` Add:
   `<meta-data android:name="apptentive_key" android:value="<YOUR_APPTENTIVE_APP_KEY>">`
   `<meta-data android:name="apptentive_signature" android:value="<YOUR_APPTENTIVE_APP_SIGNATURE>">`
3. Make sure to check whether a push came from Alchemer Mobile, and handle it yourself if `Apptentive.isApptentivePushNotification()` returns `false`.

v3.3.0

Please see the latest API Changes

Version 3.3.0 of the Alchemer Mobile Android SDK has a redesigned set of APIs for receiving and displaying push notifications from Alchemer Mobile. If you did not use Alchemer Mobile push notifications using a previous version of our SDK, you should skip this migration guide.

Major changes

Improved Push Notification Integration

- We've built simpler methods for opening a push notification that is sent from Alchemer Mobile.
- Instead of saving a push notification, opening your `MainActivity`, and then launching Alchemer Mobile from the saved push, we provide you a `PendingIntent` right when the push notification is received. You then create a `Notification` object to display to your customers, and set that `PendingIntent` into the `Notification`. When the `Notification` is opened, it will automatically launch the appropriate Alchemer Mobile Interaction.

> The changes to push require us to send the push notification in a format that is not compatible with versions of our SDK older than 3.3.0. To ensure your app continues to receive push notifications, you must migrate to these new methods when you upgrade to 3.3.0.

How to migrate

The migration process should take 5-10 minutes to complete.

1. Remove all calls to `Apptentive.setPendingPushNotification()`.

2. Remove all calls to `Apptentive.handleOpenedPushNotification()`.

3. If you were getting the push notification title and body directly from the push `Intent`, `Bundle`, or `Map` (depending on push provider), there are now convenience methods for doing so, as outline in the integration guide.

4. Follow the updated Push Notification integration instructions for your push provider.

5. There is no change to the process of setting the push token using `Apptentive.setPushNotificationIntegration()`.

v3.0.0

Please see the latest API Changes

Version 3.0.0 of the Alchemer Mobile Android SDK has extensive improvements to ease integration, improve performance and reliability, and seamlessly consume your app's styles.

Major changes

Minimum SDK Version: 14

Dropping support for ancient versions of Android ensures the Alchemer Mobile Android SDK uses that latest tools and patterns to make a great and robust experience.

Easier Integration

- We've leveraged `ActivityLifecycleCallbacks` , which means you no longer need to add Alchemer Mobile into all of your Activities. Instead, add a single line of code to your `Application.onCreate()` .
- We now use the Gradle manifest merger so that you don't need to add XML to your `AndroidManifest.xml` .
- Gradle automatically downloads AAR transitive dependencies, so you only need to add a dependency to the Alchemer Mobile Android SDK to integrate.

Inherited Styling

The SDK now inherits all styles from apps that are using an AppCompat or Material theme. This means that right out of the box, the UI of our SDK will look a lot more like your app, including colors and fonts. We also provide a way to further override our UI styles in XML.

How to migrate

Our SDK API has changed so you will need to modify your integration. Rest assured that this is a straight forward process that shouldn't take more than 20 minutes.

1. Modify your `build.gradle`
   a. In your `build.gradle` , add a dependency to Alchemer Mobile with compile `'com.apptentive:apptentive-android:3.4.0'` (or current latest version).

   b. Make sure that this dependency no longer ends in `@aar` , which will prevent gradle from downloading our transitive dependencies.

   c. You can remove any explicit transitive dependencies that you added to support Alchemer Mobile. These will be imported by gradle for you.

2. Modify your `AndroidManifest.xml`
   a. Remove everything Apptentive related. This includes permissions, Activity declarations, and the API key `<meta-data>` . tag. The new integration doesn't require modifying your manifest by default.

   b. If you added a `<meta-data>` tag for overriding our log level, you can leave this in.

3. In each Activity in your app, remove references to `ApptentiveActivity` , `ApptentiveListActivity` ,

`Apptentive.onStart()` , and `Apptentive.onStop()` . These are no longer necessary, and have been removed from our API to ensure your integration is correct.

4. In your `Application.onCreate()` , add a call to `Apptentive.register()` . This is where you will pass in your Alchemer Mobile API Key. If your app does not use an Application object yet, you must create one.

5. Various other Alchemer Mobile API method signatures have changed.
   a. Any method that formerly required an `Activity` now requires a `Context` . That means that you can call them in cases where you don't have access to an `Activity` , such as from a `BroadcastReceiver` or `Service` , and we will still be able to display our UI.

   b. Any method that formerly required a `Context` now does not require one. Integrating from the `Application` guarantees we will have access to your `ApplicationContext` , so we don't need you to pass it in anymore.

Styling the SDK

If you have specified an AppCompat or Material theme in your `AndroidManifest.xml` , our SDK inherits its look and feel from your theme. If you are using a legacy theme, such as Holo, then our SDK will not inherit your theme, and will retain our default red on white styling by default. In both cases, you can override our styles.

Please see our Android Integration Customization guide for more information.

v2.1.0

Please see the latest API Changes

v2.0.0

Please see the latest API Changes

Major Changes

Message Center

Our new Message Center is completely redesigned from the ground up, and we've made it much more configurable by modifying settings on the Alchemer Mobile dashboard. We've changed the signature of `showMessageCenter()` to return a `boolean` to tell you whether we displayed Message Center. We've also added a new method `canShowMessageCenter()` that you can use to determine whether to add a "Feedback" button to your app's UI. This is useful because Message Center configuration needs to be downloaded from our server before it can be displayed. Because of this, it may not be available for a few seconds after the first app launch.

Push Notifications

We have simplified our push notification integration by allowing your apps to receive push notifications directly from our server to your GCM enabled app. We kept support for using Parse, Airship, and Amazon SNS as well, but streamlined the existing API methods.

If you were previously using push notifications sent from Alchemer Mobile (Apptentive) in your app, see our Push Notification Integration.

> Push notifications have changed since this release. Please upgrade to v3.4.0, and follow that migration guide above.

Customer Information

We have also changed the way that you set your customer's name and email. The new methods `setPersonName()` and `setPersonEmail()` will immediately overwrite the value you or your customer have previously set, so you should only call them when you have a new value.

Unread Messages Listener

We've modified how you pass in an `UnreadMessagesListener` when you want to be notified that the number of messages waiting to be read by your customer has changed. You should replace your calls to the deprecated `setUnreadMessagesListener()` with the new `addUnreadMessagesListener()` .

Manifest file

We have removed Alchemer Mobile's `NetworkStateReceiver` from the SDK. Please make sure remove `com.apptentive.android.sdk.comm.NetworkStateReceiver` from your `AndroidManifest.xml` .

v1.7.2

Please see the latest API Changes

Major Changes

Maven / Gradle Support

Starting with this release, you can integrate our project with one line in your build.gradle. See Adding Alchemer Mobile, or see a List of Alchemer Mobile Releases.

Pushes Directly from Alchemer Mobile

If you are using a previous version of Alchemer Mobile with your push notification provider to send push notifications to your customers when you reply to their feedback, please read the following migration guide.

Push notifications have changed since this release. Please upgrade to v3.4.0, and follow that migration guide above.

How to Migrate

Parse

Parse's API has changed in their recent versions. If you are using Parse, our integration API has been significantly improved and simplified. Some older methods were removed, as they are no longer supported in new versions of the Parse SDK.

Airship

The API for Airship needed to change slightly in order to support the latest Airship SDK. Please refer to the Airship integration Instructions.

Amazon SNS

If you are using Amazon SNS for push notifications, there are no changes necessary.

—

Past this point, it will be easier to perform a fresh integration by removing everything but your `Apptentive.engage()` calls, and performing a new integration from the latest Integration reference.

v1.6.2

This version is no longer supported. Please remove your Alchemer Mobile integration and re-integrate with the latest Android Integration Reference

Please see the latest API Changes

v1.6.0

This version is no longer supported. Please remove your Alchemer Mobile integration and re-integrate with the latest Android Integration Reference

Please see the latest API Changes

Major Changes

This release refactors our source code repo to use the new Android Gradle project layout, cleans up our samples, adds Gradle support, and adds several new features.

How to Migrate

Source Code Project Structure

Our new project structure renames the Alchemer Mobile SDK directory from `apptentive-android-sdk` to simply `apptentive`. If you were previously referencing `apptentive-android-sdk` in your Eclipse or IntelliJ IDEA project, you should remove that reference and follow the instructions for Adding Alchemer Mobile.

Gradle Integration

In addition, we have added support for Gradle in this release. While we do not yet upload an AAR to a Maven repo (this began with v1.7.2), you can now use Alchemer Mobile in your gradle projects with minimal extra work. Please see the Android Integration Reference.

v1.5.0

This version is no longer supported. Please remove your Alchemer Mobile integration and re-integrate with the latest Android Integration Reference.

v1.4.0

This version is no longer supported. Please remove your Alchemer Mobile integration and re-integrate with the latest Android Integration Reference.

Related Articles