Legacy - Testing Alchemer Mobile on Android

This document will help you verify that Events and Interactions have been properly configured within your app.

Configuring Test App with Different Alchemer Mobile (formerly Apptentive) Keys

We recommend creating a new Alchemer Mobile app to test out your integration and for exploring Alchemer Mobile features.

Create a new app on your Alchemer Mobile dashboard, go to Settings -> API & Development. You can change what you pass into the `Apptentive.register()` method, but a better way is to set up a new product flavor so that your dev flavor uses different Alchemer Mobile keys than your release flavors. Set the test API key in your `AndroidManifest.xml` .

Using different flavors

First, set up Manifest Placeholders in your `build.gradle` to inject a different Alchemer Mobile App Key and Signature depending on the flavor you want to build:

```
android {
  defaultConfig {
    manifestPlaceholders = [
      APPTENTIVE_APP_KEY:"<your_release_app_key>",
      APPTENTIVE_APP_SIGNATURE: "<your_release_app_signature>"
    ];
  }
  productFlavors {
    dev {
      manifestPlaceholders = [
          APPTENTIVE_APP_KEY: "<your_dev_app_key>",
          APPTENTIVE_APP_SIGNATURE: "<your_dev_app_key>",
      ];
    }
  }
}
```

Next, add this line to your manifest:

```
<meta-data android:name="apptentive_key" android:value="${APPTENTIVE_APP_KEY}"/>
<meta-data android:name="apptentive_signature" android:value="${APPTENTIVE_APP_SIGNATURE}"/>
```

Now, you can build the dev flavor by opening Android Studio's **Build Variants** window, and changing the **Build Variant** column to *devDebug*, and then building, or by running the following command in a terminal:

```
./gradlew :app:assembleDevDebug
```

Using a test API key during development will allow you to modify Alchemer Mobile settings without fear of affecting your live app.

Testing New or Modified Interactions

In order to consistently test new or modified interactions, we recommend the following steps to make sure you see the changes you made in the dashboard:

1. Uninstall the app

2. Reinstall the app

3. Long press the app icon and open the app's info

4. Force stop the app

5. Clear the storage, then clear the cache

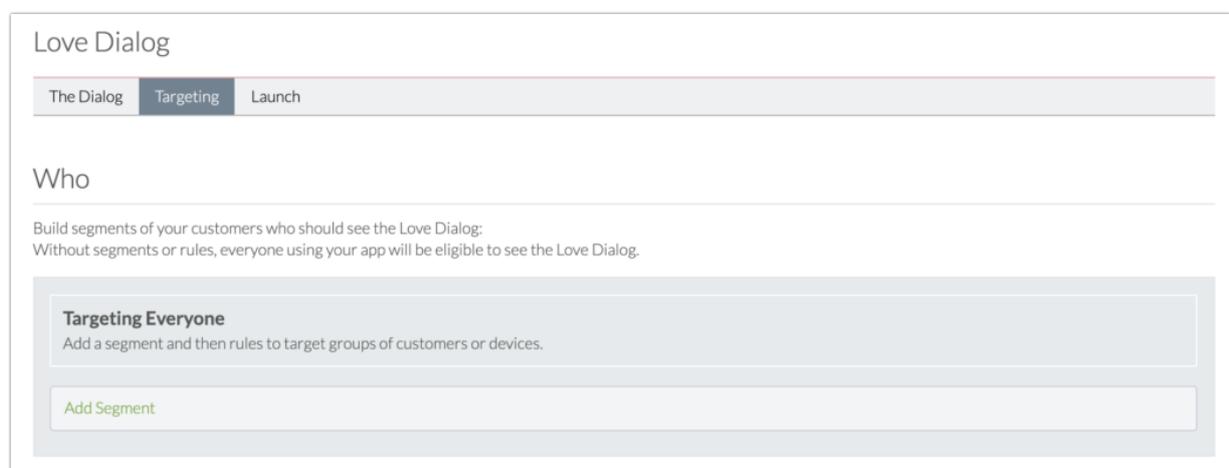6. Launch the app and test

Testing the Love Dialog and Rating Dialog

Before you begin testing, you'll need to have at least one Event integrated.

Also, if you're testing the Google Play Rating Dialog, available in Android native SDK 5.6.0 and higher, you'll need to use one of the methods outlined here.

The best way to test both of these Dialogs is to set them to run independently, also known as "decoupling" them.

Let's start by testing the Love Dialog. From this page, use the below Dashboard settings to make targeting as simple as possible.

Next, go to the Love Dialog Targeting tab and use the following settings, replacing "your_event" with an Event that you've integrated. If you don't see your event in the list, first verify that you have triggered it at least once so the server is aware of it.

Where

The Love Dialog can appear at any Event that you choose:
Pick points where your customers will be happy, engaged and won't feel disrupted by the Love Dialog.

Show the Love Dialog at any of the following Events:

your_event	✕

Add another Event:

Type to search Events...	⌄

Create New Event...

For more information about Events, view our documentation.

How Often

Wait for [1] launches of the current app version to show the dialog.

Show the dialog...
- ○ Once per person
- ● Once per person for each version of your app released

Select at least one
- ☐ Wait for [5] days after the first launch of the current app version
- ☑ Never show more than once per [90] days

[Save & Continue]

Next, we'll set up simple targeting for the Rating Dialog on this page.

If you're using an SDK that's 5.5.4 or lower, you'll be testing the Alchemer Mobile Rating Dialog. If you're using 5.6.0 or higher, you're eligible to use the in-app Google Play In-App Review Dialog. You can see examples of each, though no action is required on this page for testing.

Then, go to the Targeting tab. Simplify the targeting criteria as much as possible by using the below targeting.

Now, launch your app and engage the Where Event you selected in your targeting criteria. You should see the Love Dialog. Hit "Yes", then dismiss the Thank You Message.

Finally, engage the Where Event for the Rating Dialog. You should see the Rating Dialog.

Important: Remember, if you're testing Android native SDK version 5.6.0 or higher, you'll need to use one of these methods to see the Google Play Rating Dialog. Otherwise, you will not be able to see the dialog, and engaging the event will do nothing.

Note: If you change the Love Dialog content or logic, you will need to clear the SDK's cache by clearing your app's data or uninstalling and reinstalling it (without backups turned on in your manifest), before you will be able to see the changes in your app.

Testing Surveys

Just like with Ratings, Surveys require you to create at least one Event.

Once you have created an Event, open Surveys. There, click New Survey, and fill out your test

Survey content. After saving the first page, select Publish survey as an independent Interaction. Then, in the Where section, use the Select Event dropdown to choose your Event.

Once you have created the Survey, close and reopen the app so that it is forced to download the latest Interactions, and trigger the Event that you chose for your Survey. You should see the Survey display.

Related Articles