Integration

Android 3.6.x Android 3.2.x Android 3.3.x Android 3.4.x Android 3.5.x Android Android 3.7.x Latest version - 3.8.x Changelog

- Getting Started
- Integration
- Migration Guide
- Push Notifications
- In-App Messaging
- Inbox Messaging
- Analytics
- Deep linking
- Geofencing
- Beacons
- Advanced Features
- Advanced Use Cases
- Plugins
- Troubleshooting

Add the SDK to your app

To integrate the SDK, add the dependency for **A4SSDK** to your **app-level build.gradle** file:

app-level build.gradle

```
dependencies {
  // ...
  compile 'com.ad4screen.sdk:A4SSDK:3.8.1'
}
```

Integrate the SDK into your code

Modify your activities

To work properly Accengage SDK needs information about activities states, intents starting activities and bundles of these intents. For this reason, all of your activities should be slightly modified. There are two ways to do it:

- by adding a few lines of code into Activity methods defining the lifecycle of the activity

Modifying Activity methods

```
public class MainActivity extends AppCompatActivity {

    @Override
    protected void onNewIntent(Intent intent) {
        super.onNewIntent(intent);
        A4S.get(this).setIntent(intent);
        // ...
    }

    @Override
    protected void onResume() {
        super.onResume();
        A4S.get(this).startActivity(this);
        // ...
    }

    @Override
    protected void onPause() {
        super.onPause();
        A4S.get(this).stopActivity(this);
        // ...
    }
}
```

- by inheriting A4SActivity

Changing the parent class on A4SActivity

```
import com.ad4screen.sdk.activities.A4SActivity;

public class MyActivity extends A4SActivity {
    // ...
}
```

In case you want to use another Android standard activity, the Accengage SDK provides modified activities :

- A4SExpandableListActivity
- A4SListActivity
- A4SPreferenceActivity
- A4SNativeActivity
- A4SAccountAuthenticatorActivity

Modify you application class

If you are using the Application class, we recommend you to extend A4SApplication instead of the standard Application.

Application class

```
public class MyApplication extends A4SApplication {

    @Override
    public void onApplicationCreate() {
        // ...
    }

    @Override
    public void onApplicationTerminate() {
        // ...
    }

    @Override
    public void onApplicationLowMemory() {
        // ...
    }

    @Override
    public void onApplicationConfigurationChanged(Configuration newConfig)
{
        // ...
    }
}
```

> Since our SDK is running on its own process, your methods may be executed twice (once in your app, once in the SDK process). By extending the A4SApplication, you prevent unwanted interactions between the SDK and your App (the SDK process will not execute the code of your Application class).

In case you can not extend A4SApplication, see Sub Classing any Application Type.

Add credentials to your app

A4SSDK must be authenticated and authorized by Accengage servers. That's why you need to add application credentials (**Partner ID** and **Private key**) into **strings.xml** resource file:

strings.xml

```
<resources>
    // ...

    <string name="acc_partner_id">YOUR_PARTNER_ID</string>
    <string name="acc_private_key">YOUR_PRIVATE_KEY</string>

</resource>
```

The last two parameters presented in the code snippet are used to activate logs and disable geolocation. If you want to use geolocated In-Apps and Pushs, please check out our Geolocation section.

If you need to use different credentials, for example one pair of **Partner ID** / **Private key** per country which you would like to target, you should use localized strings: How to localize strings?

To be able to dynamically provide **Partner ID** and **Private key** with code, please check out Custom Credentials Integration sectio n.

Obfuscating with Proguard

Our SDK is already obfuscated and **Proguard configuration** is packaged in the **.aar** file.

Using Google Play Services

Our SDK plugins use Google Play Services to take advantage of certain Google-powered features like Geofencing, Geolocation, Google Advertiser ID, etc. Explore **Accengage Play Services plugins** if you would like to add these functionalities.

> We advise you to integrate the Google Advertiser ID part of the plugin to be able to identify your users through various products.

Using Splashscreen

If your app is using a Splash screen, make sure to visit our section about how to handle it: Troubleshooting Splash Screen. Otherwise, the Accengage SDK may work incorrectly. For other difficulties with our SDK, please read the Troubleshooting section for more information.

Test your integration

To test your integration, activate logs and start your application. You will see the following lines in the logcat, meaning the SDK is correctly launched:

Android Monitor

```
A4S|INFO|A4S SDK VERSION : Ax.y.z (Build : 123abc456def789hij)
A4S|ERROR|***********************************/!\***************************
********
A4S|ERROR|/!\ Logging is Enabled and must be DISABLED in production
environment /!\
A4S|ERROR|***********************************/!\***************************
********
A4S|INFO|ManifestChecker|Manifest configuration seems to be OK
```

> In order to find your device ID, please download this application: ACCInfo.
>
> You can then follow our tutorial to get to know more about how to use the application.