

## React Native (v2.1.x)

### Integrate the cmbSDK React-Native component in your App

Download the plugin via npm

```
$ npm install cmbsdk-react-native --save
```

Mostly automatic installation

```
$ react-native link cmbsdk-react-native
```

Import the component in your react-native app by adding this

- [JavaScript](#)

```
import { CMBReader, cmbComponent } from 'cmbsdk-react-native';
```

You can access all of the API methods through cmbComponent constant, and all of the enums and constants are available in the CMBReader class.

Before continuing, download the cmbSDK React-Native zip file from our download section, and see the API reference.

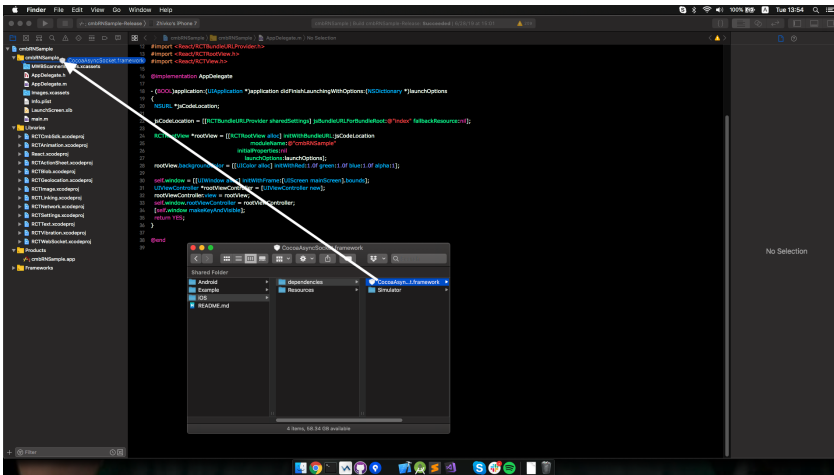
Download page: <https://cmbdn.cognex.com/download>

API reference: <https://www.npmjs.com/package/cmbsdk-react-native>

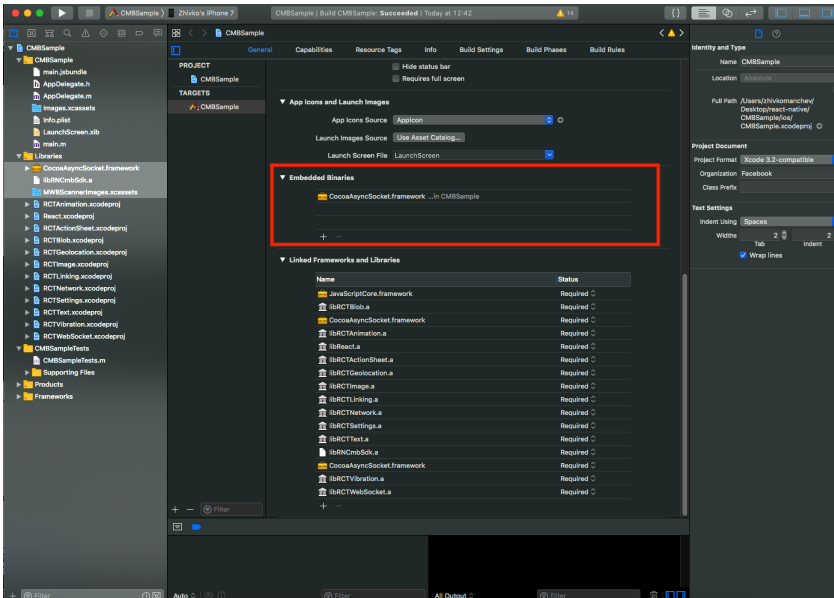
### Integrating cmbSDK iOS React-Native Component

Make sure you have downloaded the cmbSDK React-Native zip archive from our download page. From the downloaded zip file, open the iOS directory.

1. First we need to add the CocoaAsyncSocket dependency. To do that, open the "dependencies" directory and "drag and drop" the CocoaAsyncSocket.framework file in XCode. On the Adding files popup, make sure you have the "Copy items if needed" checked and in "Add to targets" your app checked. If you are using the mobile device's built in camera, do the same with the "MWBScannerImages.xcassets" file located in the iOS/Resources directory.



2. Open your targets General settings and add "CocoaAsyncSocket.framework" under "Embedded binaries"



3. In your projects info.plist file you need to add a key depending on the readerDevice type that you are using.

- If you are using the device camera for scanning, add the "NSCameraUsageDescription" key with a description on how your app will use the camera (for example: Scanning barcodes").
- If you're using a MX-1xxx device, you will need to add a "Supported external accessory protocols" array with an item "com.cognex.dmcc". You will also need to follow this [GUIDE](#) before publishing your MX-1xxx enabled app to the app store. This is ONLY required for scanning with an MX mobile terminal.

And that's it. You should be able to run your app with the cmbSDK react-native module working.

## Integrating cmbSDK Android React-Native Component

Make sure you have downloaded the cmbSDK React-Native zip archive from our download page. From the downloaded zip file, open the Android/cmbSDK-android-binary directory.

1. Open the **yourRNApp/android/app/** directory, create a new folder named "libs" if there isn't one already, and open it. Place the file "cmbSDKlib-release.aar" inside the libs folder. This aar can be found in "Android/cmbSDK-android-binary" from the downloaded zip file
2. Open the app build.gradle located in **yourRNApp/android/app** and add these lines:

- Above the dependencies add this:

```
repositories {  
  flatDir {  
    dirs 'libs'  
  }  
}
```

3. Open the build.gradle located in **yourRNApp/android** and change the minSdkVersion to 19 or above

And that's it. You should be able to run your app with the cmbSDK react-native module working.