

# Kofax mobiFlow

## React Native Bridge Developer's Guide

Version: 6.0.0

Date: 2020-10-23

The logo for KOFAX, consisting of the word "KOFAX" in a bold, blue, sans-serif font.

© 2020 Kofax. All rights reserved.

Kofax is a trademark of Kofax, Inc., registered in the U.S. and/or other countries. All other trademarks are the property of their respective owners. No part of this publication may be reproduced, stored, or transmitted in any form without the prior written permission of Kofax.

# Table of Contents

Preface.....	4
Getting help with Kofax products.....	4
Product documentation.....	5
Online documentation.....	5
Offline documentation.....	5
mobiFlow React Native Bridge for SDK.....	6
React Native sample application.....	6
Build Android React Native sample application.....	6
Build iOS React Native sample application.....	7
How to use the Bridge.....	8
Required library and framework files.....	8
iOS.....	8
Android.....	8
Licensing.....	8

# Preface

This guide includes the information you need to integrate the Kofax mobiFlow React Native Bridge into your project.

You can develop React Native applications using the Kofax mobiFlow React Native Bridge for SDK. This guide includes information on creating iOS React Native and Android React Native sample applications, that are used to demonstrate the functionality of the SDK. By adding the Bridge to your mobile application, you can use it to capture, process and extract documents and bar code data received from mobile devices.

**Note** The latest React Native version supported by the mobiFlow React Native Bridge is 0.55.4.

## Getting help with Kofax products

The [Kofax Knowledge Base](#) repository contains articles that are updated on a regular basis to keep you informed about Kofax products. We encourage you to use the Knowledge Base to obtain answers to your product questions.

To access the Kofax Knowledge Base, go to the [Kofax website](#) and select **Support** on the home page.

**Note** The Kofax Knowledge Base is optimized for use with Google Chrome, Mozilla Firefox or Microsoft Edge.

The Kofax Knowledge Base provides:

- Powerful search capabilities to help you quickly locate the information you need.  
Type your search terms or phrase into the **Search** box, and then click the search icon.
- Product information, configuration details and documentation, including release news.  
Scroll through the Kofax Knowledge Base home page to locate a product family. Then click a product family name to view a list of related articles. Please note that some product families require a valid Kofax Portal login to view related articles.
- Access to the Kofax Customer Portal (for eligible customers).  
Click the **Customer Support** link at the top of the page, and then click **Log in to the Customer Portal**.
- Access to the Kofax Partner Portal (for eligible partners).  
Click the **Partner Support** link at the top of the page, and then click **Log in to the Partner Portal**.
- Access to Kofax support commitments, lifecycle policies, electronic fulfillment details, and self-service tools.  
Scroll to the **General Support** section, click **Support Details**, and then select the appropriate tab.

## Product documentation

By default, the Kofax mobiFlow documentation is available online. However, if necessary, you can download the documentation to use offline.

### Online documentation

The product documentation for Kofax mobiFlow 6.0.0 is available at the following location:

[https://docshield.kofax.com/Portal/Products/en\\_US/mobiFlow/6.0.0-tss0pu9zau/mobiFlow.htm](https://docshield.kofax.com/Portal/Products/en_US/mobiFlow/6.0.0-tss0pu9zau/mobiFlow.htm)

### Offline documentation

To access the documentation offline, download the documentation .zip files from the [Kofax Fulfillment Site](#) and extract them on a local drive available to your users.

# mobiFlow React Native Bridge for SDK

React Native is an open source mobile application created by Facebook. The Kofax mobiFlow React Native Bridge for SDK is used to create mobile applications that you can use to capture and process images and bar code data received from mobile devices.

The Bridge exposes much of the mobiFlow SDK functionality within a React Native application. The bridge calls existing SDK methods and sends the response back to the JavaScript. The Bridge contains native methods that are exposed to the Javascript.

The React Native related files provided with the Bridge are described in the following table.

File Name	Description
MobiFlowModule.java MobiFlowReactPackage.java	These are the native parts of the Android Bridge. These parts are responsible for interacting with native libraries.
ReactNativeMobiFlow.h ReactNativeMobiFlow.mm	These are the native parts of the iOS Bridge. These parts are responsible for interacting with native libraries.

## React Native sample application

A sample React Native application demonstrates the Bridge is available in the `Hybrid/ReactNative` folder. This sample application can be used on iOS and Android devices.

The sample application demonstrates the capture, process and extraction features with the **Check** components.

When user taps on the **Scan** button, the application launches the check capture view controller, and after capturing results are displayed.

The Check component has been exposed for this sample application. You can capture Full Page, Bill, Custom, Credit Card and Passports components by using Kofax mobiFlow React Native Bridge. To launch the other components you need to make changes in the code.

## Build Android React Native sample application

1. To open ReactNative, run the following command in the Terminal or Command Prompt:  

```
cd "Hybrid/ReactNative/ path"
```
2. To install the dependencies, run the following command:  

```
npm install
```
3. In `Hybrid/ReactNative/android/app/src/main`, if **assets** folder exists, then remove the contents from it. If it does not exist, create a folder with the name **assets**.

4. Update the license in `Hybrid/ReactNative/App.js` file.
5. To load the contents to the **assets** folder, run the following command:

```
node node_modules/react-native/local-cli/cli.js bundle --platform android --dev
false
--entry-file index.js --bundle-output android/app/src/main/assets/
index.android.bundle
--assets-dest android/app/src/main/res
```

6. Copy the `mobiFlow.Android-release.aar` file from the `KofaxmobiFlowSDK (Android/aars)` folder to the `Hybrid/ ReactNative/android/app/libs` folder.
7. Open the Android project (`ReactNative/android`) in Android Studio.
8. Run the application.

## Build iOS React Native sample application

1. To open ReactNative, run the following command in the Terminal or Command Prompt:

```
cd "Hybrid/ReactNative path"
```

2. To install the dependencies, run the following command:

```
npm install
```

3. Update the license in the `ReactNative/App.js` file.

4. To update jsbundle, run the following command:

```
node node_modules/react-native/local-cli/cli.js bundle --entry-file index.js --
platform ios --dev false
--bundle-output ios/main.jsbundle --assets-dest ios
```

5. Unzip the Frameworks from `\iOS\mobiFlow\Frameworks`.
6. Copy `opencv2.framework` and `KofaxmobiFlowWidget.framework` from the `KofaxmobiFlowSDK (iOS/mobiflow/Frameworks)` folder to the `ReactNative/ios/MobiFlowSDK` folder.
7. Copy the Resources folder from the `KofaxmobiFlowSDK (iOS/mobiflow)` folder to the `ReactNative/ios/MobiFlowSDK` folder.
8. Open the `ReactNativeMobiFlow.xcworkspace` project in `ReactNative/ios`.
9. Change the code signing development team for the both `ReactNativeMobiFlow` and `ReactNativeMobiFlowTests` targets.
10. If you get the error "config.h not found in demangle.h", remove the **third-party** folder from the `ReactNative/node_modules/react-native` folder.
11. If you get the error "unknown argument type '\_\_attribute\_\_' in method -[RCTAppState getCurrentAppState:error:] Extend RCTConvert to support this type", update the method `RCTParseUnused` in the class `ReactNative/node_modules/react-native/React/Base/RCTModuleMethod.mm` with following code:

```
{
return RCTReadString(input, "__unused") || RCTReadString(input,
"__attribute__((__unused__))") || RCTReadString(input,
"__attribute__((unused))");
}
```

12. Run the application.

## How to use the Bridge

You can use the Bridge with the existing application or with a newly created application. You need make some changes to the project. For instructions, go to `\Hybrid\ReactNative` folder and open the `ReadMe.md` file.

## Required library and framework files

To use MobiFlow SDK with your application, refer to the *Kofax mobiFlow SDK Developer's Guide* for specific library and frameworks.

### iOS

For an iOS project, the following frameworks are required:

- `KofaxmobiFlowWidget.framework`
- `opencv2.framework`
- Localized string files and bundle in the Resources folder

### Android

For an Android project, the following library is required:

- `mobiFlow.Android-release.aar`

## Licensing

When you use the Bridge in the sample application, you need to pass your mobiFlow SDK license to the Bridge.

For the sample application, the license is set in the parameters of the scan method, which you need to do in the `App.js` file.

In the sample application directory modify `ReactNative/App.js` to insert your license.