Payrix Mobile

Software Development Kit (SDK)

iOS Developer Guide

Version 3.0.15

Revision Date: 08.24.2023

Payrix

TABLE OF CONTENTS

Overview

The Payrix Mobile SDK for iOS allows a client to create a full mobile payment app that handles card and transaction processing against a designated Payrix payment gateway. Version 3 of the SDK has been greatly simplified to speed development efforts. In addition, the SDK is now EMV certified, which allows the client to effectively create EMV certified payment apps.

The Payrix Mobile SDK version 3 will no longer use CocoaPods. We employ a standard GIT download while we evaluate the best distribution / dependency solution.

Requirements

The Payrix Mobile SDK was designed and developed to leverage the following requirements:

- iOS 12 or Later
- XCode IDE
- Bluetooth and Audio Card Readers (or Manual Card Entry)
- Network accessibility (WIFI or Cellular)

Setting Up the SDK

Integrating the PayrixSDK into your Payment App

1. Download the Payrix Mobile SDKs Version 3.0 folder located at: https://gitlab.com/payrix/public/payrix_mobile_sdk_v3/ios_sdk_v3

2. After you unzip the download file you will find 4 files of interest:
    a. This Developer's Guide for iOS
    b. The PayrixSDK.framework file
    c. A folder titled: Bitcode Enabled Framework
    d. The PayrixSDK3Demo folder

3. Adding the Payrix SDK to your App:
    a. Within your project folder there should be a subfolder named: Frameworks. If it does not exist, then you can create it.
    b. Next drag the PayrixSDK.framework file to the Frameworks folder. If your app requires a Bitcode Enabled framework, then use the PayrixSDK.framework file located in the Bitcode Enabled Framework folder.
    c. Select Copy on the popup.
    d. Next with the Project level selected, select General in the main pane.
    e. Scroll down to the section titled "Frameworks, Libraries, and Embedded Content"
    f. You should see PayrixSDK.framework listed. Under the caption "Embed", use the pulldown to select "Embed & Sign"

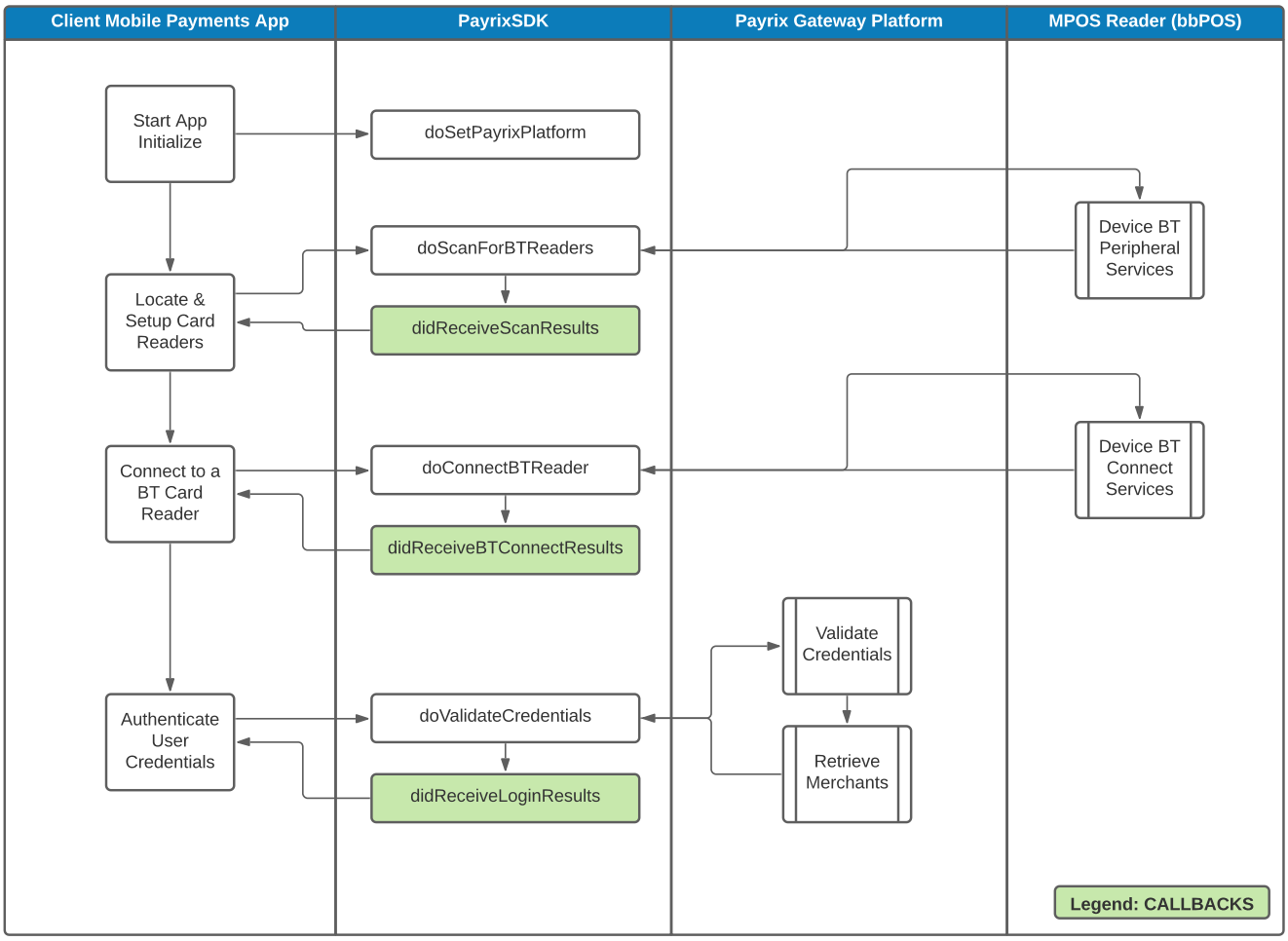
4. You are now ready to use the new PayrixSDK in your App.

Using the Payrix Mobile SDK

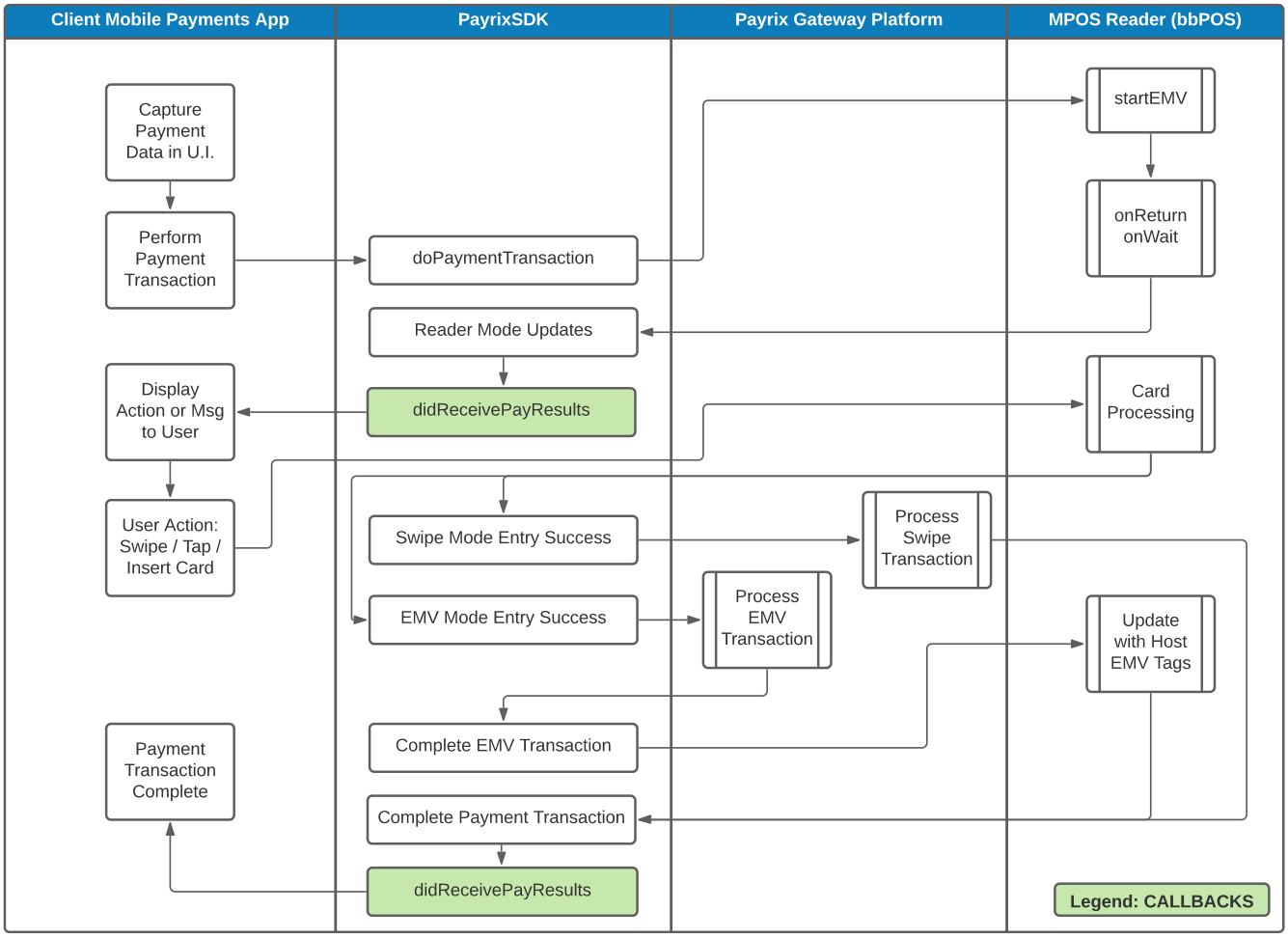
The PayrixSDK version 3 is now contained in one framework (PayrixSDK) that handles Bluetooth card readers, Bluetooth communications, and payment transaction processing to a Payrix gateway.

PayrixSDK General Process Flow

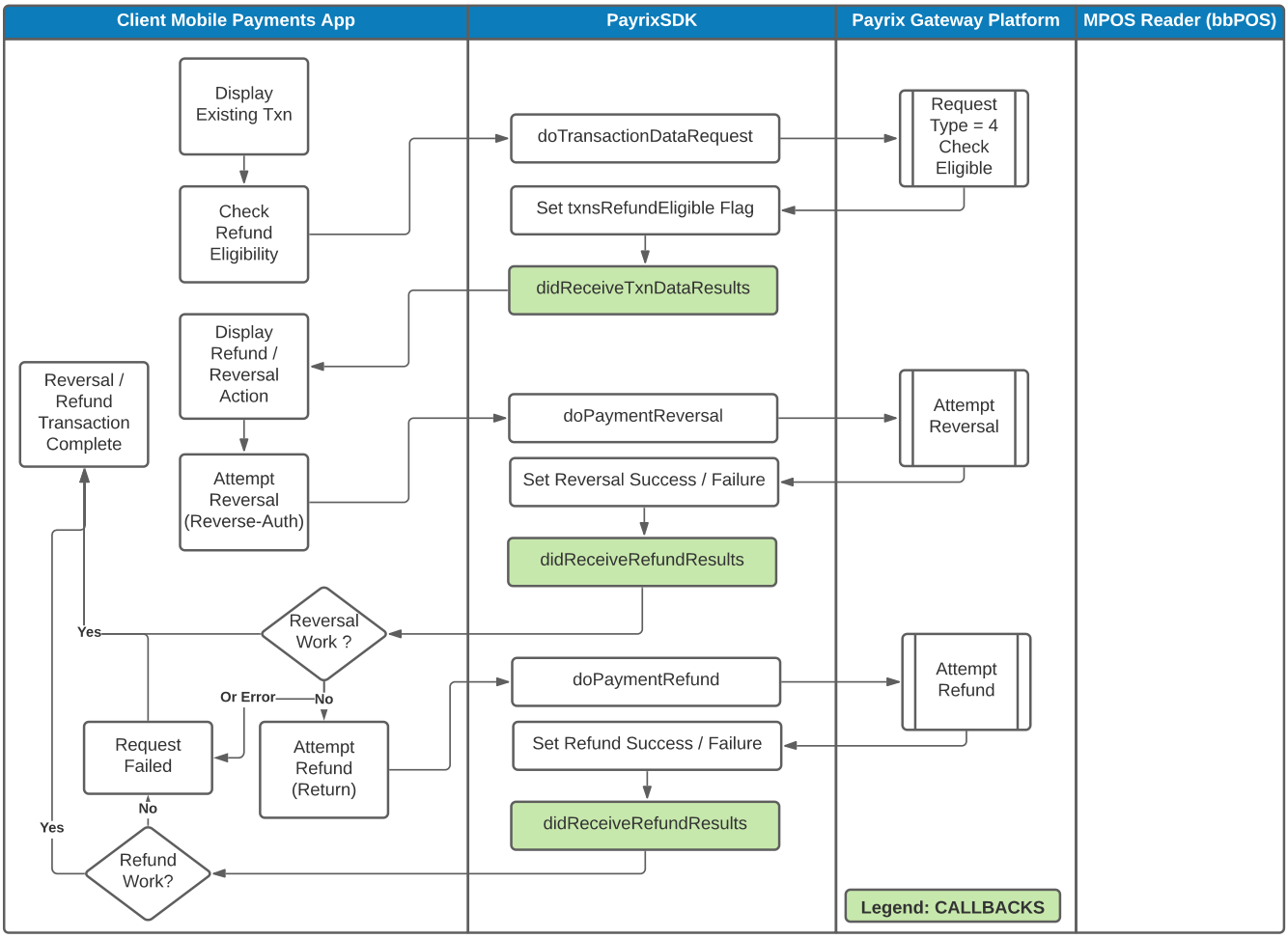
Connection and Authentication



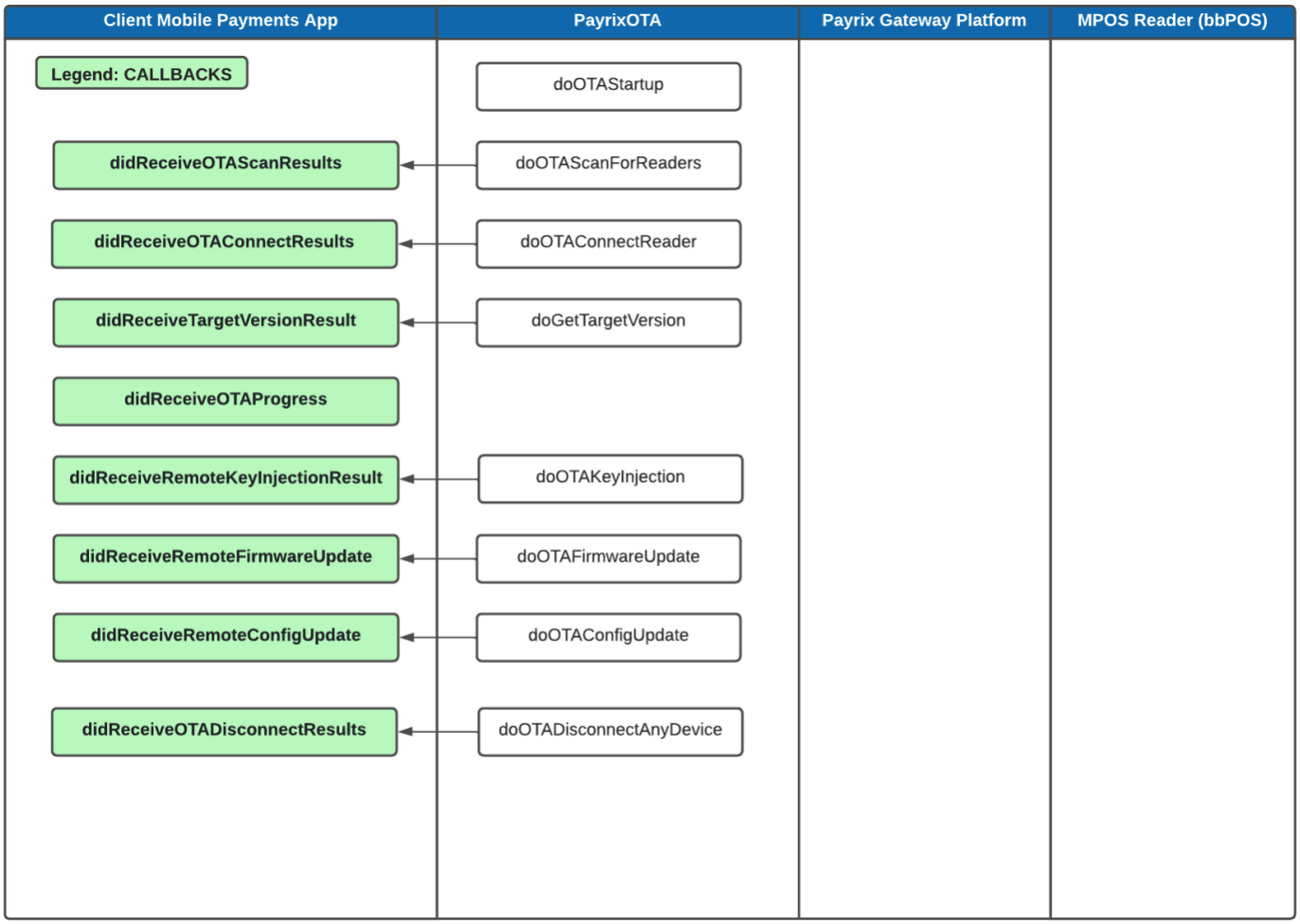
Payment Transaction



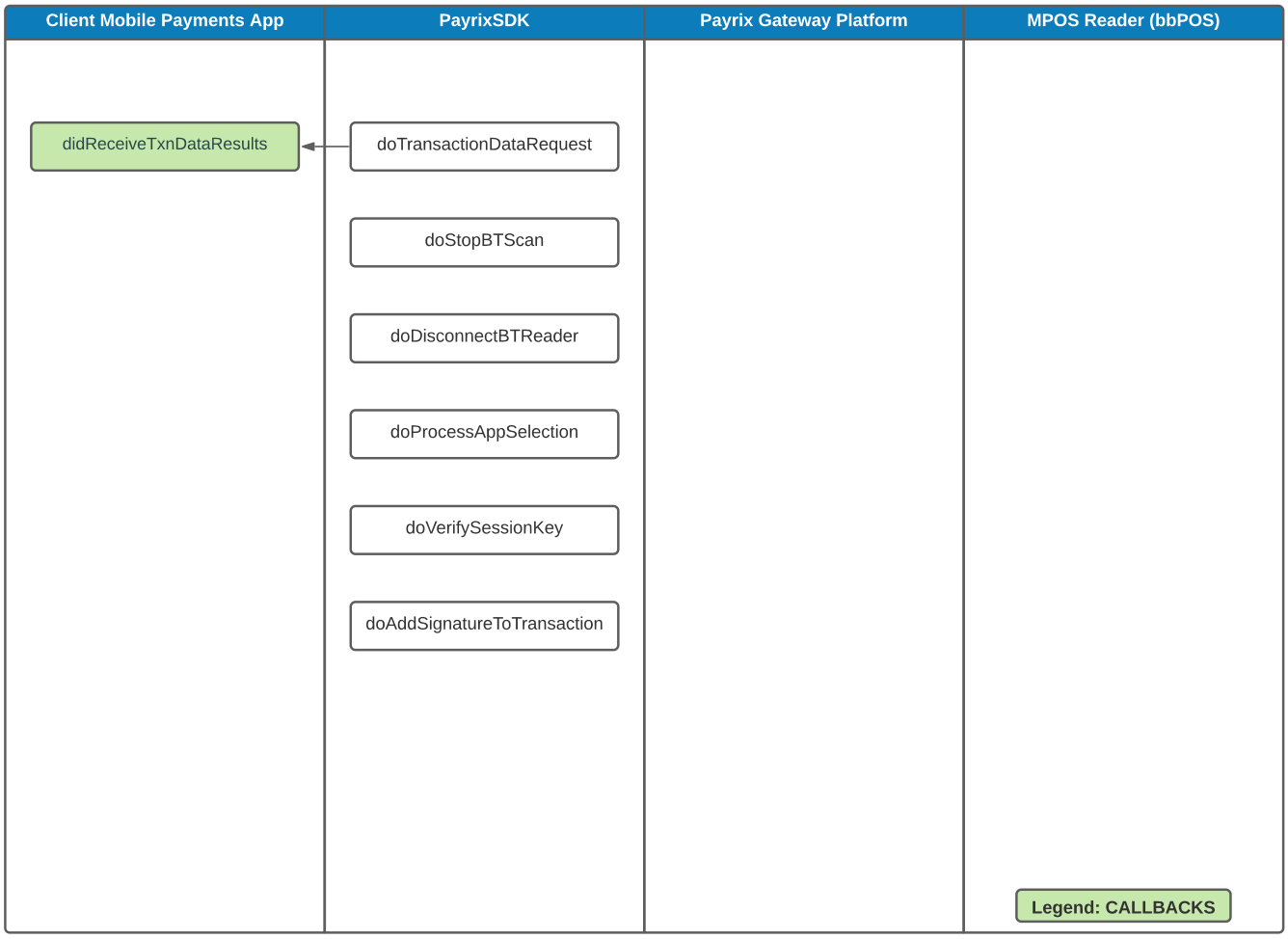
Reverse-Auth (Reversal) / Refund Transaction



Over The Air (OTA) Device Configuration Functions



General Functions



Note that for the classes and methods shown Xcode help is available. By holding the Option Key and clicking on the Class / Method a help popup screen will provide a description, the input parameters (if any), and the returned values. Most complex methods where the response may not be immediate may return the results via a callback and will be shown below.

PayrixSDK Methods (all the initializer is changed to PayrixSDKMaster)

Class	Method / Property (p)	Purpose	Callback
Payrix SDK Instantiation Commands			
PayrixSDK	sharedInstance (p)	Access to PayrixSDK singleton class	delegate: Set to the class that will handle method specific callbacks.
	doSetPayrixPlatform	Establishes the base working environment for the app. Parameters: • Payrix Gateway URL • Sandbox (On/Off) • Device Manufacturer from BT devices object	N/A
Bluetooth Reader SDK Commands			
PayrixSDK	doScanForBTReaders	Scans for available Bluetooth readers.	didReceiveScanResults: • Success (True /False) • Message • Array of PayDevice[1] objects
	doConnectBTReader	Connects a Bluetooth device to the App for Payment Processing. Parameters: • PayDevice object	didReceiveBTConnectResults: • Success (True/False) • Reader Device
	doStopBTScan	Stops a BT Scan for Readers	N/A
	doDisconnectBTReader	Disconnects a currently connected BT Reader	didReceiveBTDisconnectResults: • Success (True/False)
SDK Authentication Commands			
PayrixSDK	doValidateCredentials	Accepts Parameters to Validate Credentials: • UserID (String) • Password (String)	didReceiveLoginResults: • Success (True/False) • Session / API Key • Array of PayMerchant[2] objects • Message
	doVerifySessionKey	Accepts Parameter to verify the session key is still valid: • SessionKey (String)	N/A Returns a Boolean (True/False) True = Valid False = Invalid
	doGetSingleMerchant	Accepts Parameters to get the single merchant: • merchantID (String) • sessionKey (String)	didReceiveSingleMerchantRetrievalResults: • fetchSuccess (True/False) • theMerchant - PayMerchant object • theMessage
SDK Payment Command			
PayrixSDK	doPaymentTransaction	This command accepts a payment request object as a parameter and performs all of the activities needed to process a transaction from	didReceivePayResults: • ResponseType ○ 0 = Transaction Complete ○ 1 = Action Message ○ 2 = Info-Only Message ○ 3 = Reserved

		card processing to Payrix transaction processing. • PayRequest object	○ 4 = App Selection Needed ○ 5 = Sending Final EMV Data ○ 9 = Error Occurred • ActionMsg: Requires user action such as “Insert / Tap / Swipe Card” • InfoMsg: A general message such as “Processing..” • PayResponse object with the data elements of a processed transaction.
PayrixSDKMaster	doCancelCheckCard	As soon as transaction is started but any card operation is not performed, Swipe, Tap or Insert, then this can be used to cancel checking the card	didReceivePayResults: • ResponseType ○ 0 = Transaction Complete ○ 1 = Action Message ○ 2 = Info-Only Message ○ 3 = Reserved ○ 4 = App Selection Needed ○ 5 = Sending Final EMV Data ○ 9 = Error Occurred • ActionMsg: Requires user action such as “Insert / Tap / Swipe Card” • InfoMsg: A general message such as “Processing..”
Optional Command			
PayrixSDKMaster	doDebugEnable	Enable the debug log	N/A
PayrixSDKMaster	doSetConsoleToFile	N/A	N/A
PayrixSDKMaster	doWriteConsoleFile	N/A	N/A
PayrixSDKMaster	doGetDeviceInfo	Getting info of connected device, method only works if any device is connected	didReceiveDeviceInfo: • deviceInfo: [AnyHashable : Any]!)
PayrixSDKMaster	doInitiatePaymentRefund	This method is the entry point for the execution of a Payment Refund. The payment refund steps are as follows	N/A
	doManualEntryTrans		didReceivePayResults: • ResponseType ○ 0 = Transaction Complete ○ 1 = Action Message ○ 2 = Info-Only Message ○ 3 = Reserved ○ 4 = App Selection Needed ○ 5 = Sending Final EMV Data ○ 9 = Error Occurred • ActionMsg: Requires user action such as “Insert / Tap / Swipe Card” • InfoMsg: A general message such as “Processing..”
	doRetrieveRefundReadyTxns		
	doGetSubsequentTransactions		

Class	Method / Property (p)	Purpose	Callback
SDK Transaction Data Command			
PayrixSDK	doTransactionDataRequest	Request Type: 1. Retrieve Specific Txn 2. Retrieve All Txns for a Merchant 3. Retrieve Related (Subsequent) Txns for a Txns 4. Check if Txns is Refund Eligible	didReceiveTxnDataResults: • Success (True/False) • Response Code = Request Type • Txn Message: Info Message • Txn Data Response Object
SDK Reversal / Refund Commands			
PayrixSDK	doPaymentReversal	The method provides Reverse-Auth (Reversal) functionality. Parameter Passed: • Refund Request Object	didReceiveRefundResults: • Success (True /False) • Response Codes: ○ 0 = Refund / Reversal Successful ○ 1 = Refund Declined ○ 2 = Reversal Declined ○ 3 = Reserved ○ 4 = Reserved ○ 5 = Invalid Amt Requested ○ 6 = Reserved ○ 7 = Reserved ○ 8 = Not Used ○ 9 = Unexpected Error (See Msg) • Refund Message • Refund Response Object
	doPaymentRefund	The method provides Refund (Return) functionality. Parameter Passed: • Refund Request Object	didReceiveRefundResults: • Success (True /False) • Response Codes: ○ 0 = Refund / Reversal Successful ○ 1 = Refund Declined ○ 2 = Reversal Declined ○ 3 = Reserved ○ 4 = Reserved ○ 5 = Invalid Amt Requested ○ 6 = Reserved ○ 7 = Reserved ○ 8 = Not Used ○ 9 = Unexpected Error (See Msg) • Refund Message • Refund Response Object

Class	Method / Property (p)	Purpose	Callback
General Purpose SDK Commands			
PayrixSDK	doProcessAppSelection	EMV cards contains "Application" logic to deal with a variety of card processing services including fraud detection. During the process an EMV device may ask which App to use. The SDK will prompt in the "didReceivePayResults" callback. The selection response is handled in this method call. The parameters are: • App Index (Index position in array of Apps sent to select from) • App Name (Name of App selected)	N/A
	doAddSignatureToTransaction	Store the signature of customer on Payrix platform for dispute resolution. Parameters: • Session Key • Original Txn ID • Signature in a Base64 Encoded String	N/A Returns: • Success (True / False) • Error Messages Array (if Required)

Object Classes

Important Note:
The following objects are subject to change and new ones added before this document may be updated. The actual SDK and the inline documentation is the best reference. Payrix is committed to maintaining this document as well, but the SDK is the best and most accurate reflection of the objects.

PayDevice

Element Name	Data Type	Comment
readerDevice	String	Device name
deviceUUID	String	Bluetooth UUID reader device
deviceManfg	String	Supported PayCardSupportedReader Manufacturer
deviceSerial	String	Connection Code / Serial Number

PayEMVTags

Element Name	Data Type	Comment
AID_4F	String	Industry Name [underscore] Tag ID
EMVChipInd	String	Swipe | Manual Entry | Chip
AIDName_9F12	String	Industry Name [underscore] Tag ID
PINStmt	String	
TVRCVR_95	String	Industry Name [underscore] Tag ID
ApprovedDeclined	String	
AuthApprovalCode	String	
TSI_9B	String	Industry Name [underscore] Tag ID
CryptoCert_9F26	String	Industry Name [underscore] Tag ID

PayMerchant

Element Name	Data Type	Comment
merchantID	String	
merchantDBA	String	
entityID	String	
addressLine1	String	
addressLine2	String	
city	String	
stateprovince	String	
postalCodezip	String	
countryCode	String	

PayRequest

Element Name	Data Type	Comment
payTotalAmt	Double	The Total amount (includes Tax and Tip)
payTaxAmt	Double	A Tax amount in decimal form (6.75)
payTipAmt	Double	A Tip amount in decimal form (1.05)
payCurrencyCode	String	A Universal Currency Code (Ex: USD)
payHostURL	String	Payrix Host URL
payDeviceMode	PaySharedAttributes. PayDeviceMode	Swipe, Insert, Tap, or some combination (See definition for full list)
paySessionKey	String	A valid session key (or API Key) from a successful credentials login
payrixMerchantID	String	A valid Payrix Merchant ID
payrixSandoxDemoMode	Bool	An Indicator: True = Sandbox Demo mode; False = Live - Production Mode
payManualEntry	Bool	An Indicator: True = Manual Card Data Entry; False = Card Reader Capture (Default)
order	String	The identifier of the Order associated with this Transaction. This field is stored as a text string and must be between 0 and 200 characters long.
Following Fields Required for Manual Entry Only		
payAmount	Double	The Amount (Cost) without Tax or Tip
payTaxPercent	Double	A Tax Rate (Percent)
payTipPercent	Int	A Tip Percent Amount
payCardHolder	String	The Card Holder Name
payCCNumber	String	The Card Number
payCardType	PaySharedAttributes .CCType	The Card Type (Brand)
payCardCVV	String	The CVV Security Code
payCardExp	String	The Card Expiration Date
payOrigin	PaySharedAttributes .PayTxnOrigin	The Transaction Origin: ECommerce, CreditCardTerminal, etc
payAddress1	String	
payAddress2	String	
payCity	String	
payStateProvince	String	
payPostalCodeZip	String	The Card Holder Billing Zip Code
Optional Fields Used in Response to Card Reader Request		
appSelectionIndex	Int	Response to App Selection Request from Card Reader Device (Index number from passed AppSelection array)
appSelectionName	String	Response to App Selection Request from Card Reader Device (Card App Name from passed AppSelection array)

PayResponse

Element Name	Data Type	Comment
merchantID	String	
merchantDBA	String	
transactionID	String	
amount	Double	
ccNumber	String	
ccCardType	PaySharedAttributes .CCType	
ccName	String	
ccEXP	String	
ccCVV	String	
ccZip	String	
tipAbsoluteAmount	Double	
taxPercentage	Double	
signature	UIImage	
signatureBase64Encoded	String	
errorMessages	[String]	
receiptAID_4F	String	
receiptEMVChipInd	String	Swipe | Manual Entry | Chip
receiptAIDName_9F12	String	
receiptPINStmt	String	
receiptTVRCVR_95	String	
receiptApprovedDeclined	String	
receiptAuthApprovalCode	String	
receiptTSI_9B	String	
receiptCryptoCert_9F26	String	
tag8ARespCode	String	
tag91AuthData	String	
tag71Script1	String	
tag72Script2	String	
appSelection	String	
tenderType	String	Tender Type = Credit or Debit
posEntryMode	String	POS Entry Mode: Swipe, Chip (Insert), Contactless
receiptAppLabel	String	
receiptSignLineRequired	Bool	
receiptAddressLine1	String	
receiptAddressLine2	String	
receiptCity	String	
receiptStateprovince	String	
receiptPostalCodezip	String	

receiptCountryCode	String	
receiptTerminal	String	
receiptCardBrandName	String	
finalEMVTags	[AnyHashable:Any]	String?
payTxn	PayCoreTransResponse	
originalPayRequest	PayRequest	
appSelectionIndex	Int?	Response to App Selection Request from Card Reader Device (Index number from passed AppSelection array)
appSelectionName	String?	Response to App Selection Request from Card Reader Device (Card App Name from passed AppSelection array)
debugSDKData	[String: String]?	Debug info from Payrix SDK

The Payrix SDK Demo App

The demo app provides assistance on how to perform the most important steps of a payment transactions using a Bluetooth card reader. In addition, the app validates that the software is installed properly, and the solution is in working order.

The Payrix Demo App is a fully functional app and was used in the EMV certification of the Payrix SDK. To install the app just copy it from the Payrix SDK Version 3.0 folder and open it in Xcode and you should be able to compile and run the product on a IOS device.

Feedback and Support

To provide feedback on this document and any of the specifications contained within, contact us at: support@payrix.com

Document Revision History

Revision Date	Author	Revision Description
02.05.2021	Steven Sykes Mobile Design & Engineering	Original Document
12.10.2021	Steven Sykes Sr. Manager | Client Platforms	Revised to support EMV Certified version of Payrix SDK.
02.04.2022	Steven Sykes Sr. Manager | Client Platforms	Revised to add installation information
02.20.2022	Steven Sykes Sr. Manager | Client Platforms	Revised to add OTA Info.
03.21.2022	Steven Sykes Sr. Manager | Client Platforms	Revised to support Bitcode enabled version of the SDK.
04.21.2023	Prakash Kotwal Sr. iOS Engineer	Resolved the issue for crash occurring when Payrix SDK finds the devices that has same UUID as BBPos but has no name on it.
08.24.2023	Prakash Kotwal Sr. iOS Engineer	1. Successfully addressed the concern of not receiving target version for CHB2F Devices. 2. Introduced a new setDebugLog() method to facilitate enhanced logging in OTA operations. 3. Eliminated Bitcode Support for improved compatibility. 4. The SDK exclusively supports Xcode14 and higher versions.
