

# PayTabs iOS SDK Integration Guide

INTEGRATION GUIDE  
VERSION 3.0

## Table of Contents

1	Overview	5
1.1	Test Merchant Account v/s Live Merchant Account	5
1.2	Target Audience	5
1.3	Assistance	5
1.4	Documentation Feedback	5
2	Importing the SDK iOS Project	6
3	Integrating the SDK with your application	8
3.1	SDK Parameters	8
3.2	Integration steps and guidelines	12
3.3	SDK Response	13
3.4	Language Configuration	14
3.5	Important Notes	15
4	Verify Transaction	15
4.1	Request	15
4.2	Response	16
4.3	Response Codes	17
5	PayTabs IPN (Instant Payment Notification)	17
5.1	How IPN Works	17
5.2	Set up your IPN Listener	18
5.3	IPN Events	18
5.4	Contents / Data of the IPN	19
5.5	Response Codes	20
5.6	Sample IPN Response	20
5.6.1	IPN sample response for successful SDK transactions	20
5.6.2	IPN sample response for rejected SDK transactions	21
5.6.3	IPN sample response for rejected SDK transactions by fraud monitoring	21
5.6.4	IPN sample response for force accepted SDK transactions	22
5.6.5	IPN sample response for refunded SDK transactions	22
6	Refund Transactions	23
6.1	Request	23
6.2	Response	24
6.3	Response Codes	24
7	Transactions Reports	26

7.1	Request	26
7.2	Response	27
7.3	Error Codes	27
8	Tokenization in SDK	28
9	Test Credit Card Numbers	29
10	Screenshots of the application	30
10.1	PayTabs Mobile Payment Page	30
10.2	Successful Payment Result Page	30
11	Frequently Asked Questions	31

## Revision History

Rev	Description of Change	Date
1.0	- New Document	15/01/2015
1.1	- Formatted Document - Added documentation feedback email address. - Added important note for secret key and spaces. - Updated credit card expiry dates. - Added code on how to view response codes	29/06/2015
1.2	- Added the different response codes and their description. - Added important note on simulators.	16/08/2015
1.3	- General Document Enhancements	21/01/2016
1.4	- Updated Header files screenshot	25/01/2016
1.5	- Add screenshots of the application	02/02/2016
2.0	- Update SDK to Version 2.0 - Added section "Frequently Asked Questions"	21/03/2016
2.1	- Added Tokenization feature	18/07/2016
2.2	- Added parameter card_last_four_digits in verify transaction response - Added FAQ about hybrid frameworks & Swift Code	27/07/2016
2.3	-Added FAQs related to iOS SDK -Added Sample Response Codes for Paytabs IPN	29/06/2017
2.4	-Remove screenshots and details related to Scan To Pay	28/08/2017
3.0	- A complete revamped of SDK - New mechanism of initiating and integrating SDK - Callbacks added to handle transaction responses - SDK is upgraded to static iOS framework with encrypted resources bundle - Added validations to minimize errors	12/12/2017

## 1 Overview

At PayTabs, we are committed to providing the most secure, reliable and user-friendly payment processing solutions. By partnering with us for your payment processing needs, you can be confident that payment transactions will be processed quickly and efficiently, and your customers' information will be safe.

You'll enjoy our iOS SDK's easy set-up and easy integration into the iPhone applications you develop. Our two-level fraud protection system protects transaction data throughout the payment process. With PayTabs as your payment processor, you'll benefit from the fastest, most secure, and cost-effective payment processing system in the industry.

It's important to note that the PayTabs APIs and SDKs are constantly evolving, and updates are usually done every quarter, if required. PayTabs will intimate any major change, so it's important that you check your merchant dashboard notifications or e-mail, frequently to be updated.

### 1.1 Test Merchant Account v/s Live Merchant Account

PayTabs does not offer an explicit Sandbox / Testing environment. But using a demo account, you can freely test all the functionalities and integrate our API, SDK and Plugin without any hassle. In order to move to LIVE PRODUCTION environment, you will need to activate your LIVE MERCHANT ACCOUNT by clicking on [GO LIVE](#) through your PayTabs Merchant Account. PayTabs requires some personal information, documents and other details related to your business to validate your business. Once your Merchant Account is activated, you will have been moved to the LIVE PRODUCTION environment.

### 1.2 Target Audience

This document is intended for developers who wish to develop iPhone applications that interact with PayTabs.com

### 1.3 Assistance

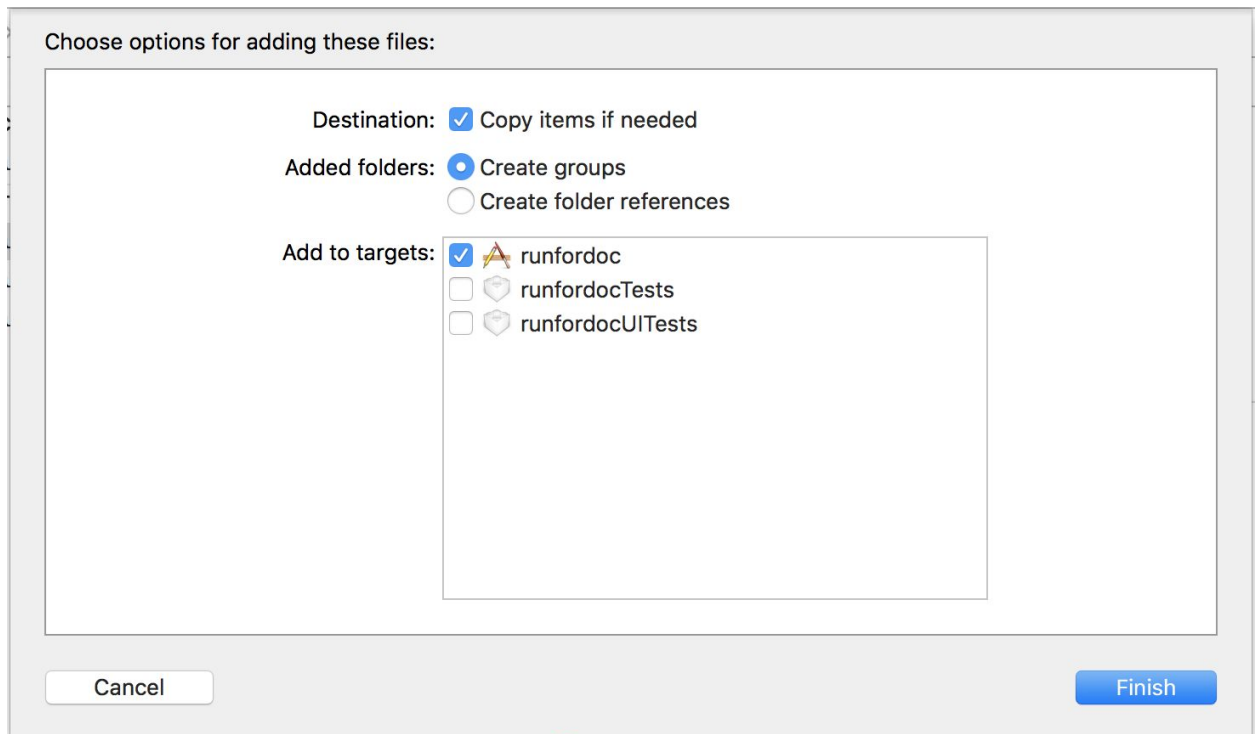
If you require assistance, please check out our [Frequently Asked Questions](#) or chat live with our support staff or email us at [support@paytabs.com](mailto:support@paytabs.com).

### 1.4 Documentation Feedback

Help us improve the PayTabs iOS SDK Integration Guide, by sending your feedback to [support@paytabs.com](mailto:support@paytabs.com).

## 2 Importing the SDK iOS Project

1. Extract the **PayTabs iOS SDK.zip** file which contains two folders, **Framework** and **Resources**.
2. **Framework** folder contains *paytabs-ios.framework* (static iOS framework) and *Resources.bundle* (includes interface files, images, local files, etc)
3. Add *paytabs-ios.framework* into your workspace by dragging and with **Copy items if needed** checked.



4. Perform the same steps for *Resources.bundle*.
5. Add following pod dependencies in your podfile, don't add those which are already in the podfile. Make sure you have uncommented **platform:ios, 'x.x'**. Your podfile

should look like,

```
# Uncomment the next line to define a global platform for your project
# platform :ios, '9.0'

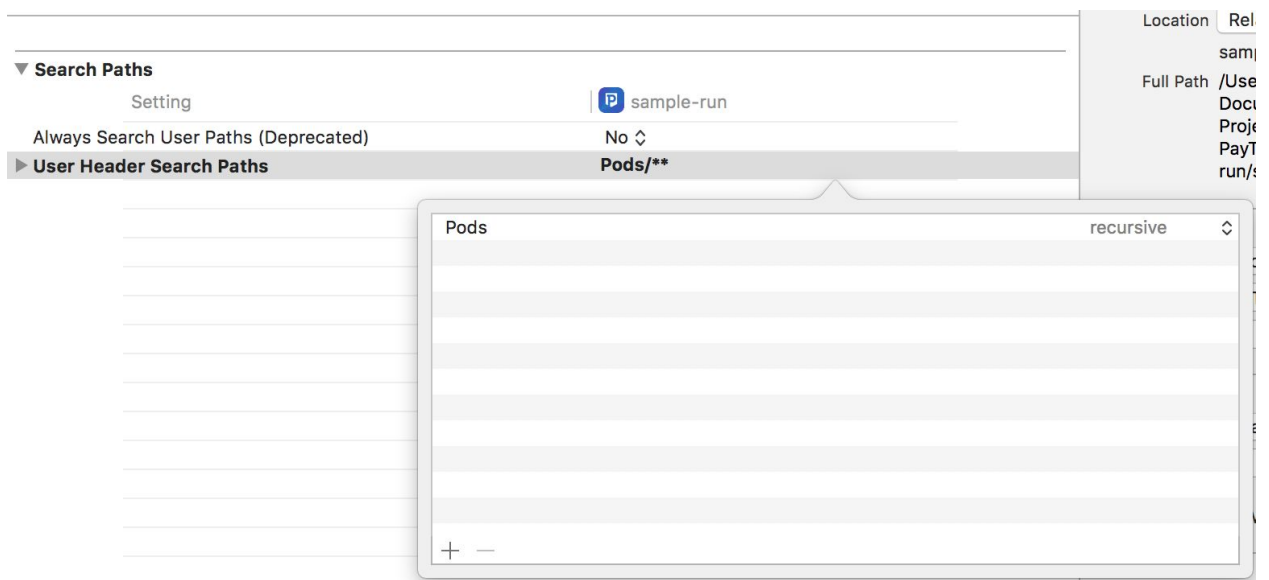
target 'sample-run' do
  # Comment the next line if you're not using Swift and don't want to use dynamic frameworks
  # use_frameworks!

  # Pods for sample-run
  pod 'BIObjCHelpers'
  pod 'AFNetworking'
  pod 'Mantle'
  pod 'Reachability'
  pod 'SDWebImage'
  pod 'DGActivityIndicatorView'
  pod 'Lockbox'
  pod 'YLGIFImage'
  pod 'SBJson'
  pod 'PINCache'
  pod 'IQKeyboardManager'

  target 'sample-runTests' do
    inherit! :search_paths
    # Pods for testing
  end

  target 'sample-runUITests' do
    inherit! :search_paths
    # Pods for testing
  end
end
```

6. Please add **Pods** in **User Header Search Paths** with **Recursive** option within your app target.



7. If your application programming language is **Swift**, you will need to have **Bridging Header** for importing `#import <paytabs-ios/paytabs_ios.h>`. Also, give your bridging header path against **Objective-C Bridging Header** in your app target's

build settings.

```
//  
// sample-run-Bridging-Header.h  
// sample-run  
//  
// Created by PayTabs LLC on 10/5/17.  
// Copyright © 2017 PayTabs LLC. All rights reserved.  
//  
  
#ifndef sample_run_Bridging_Header_h  
#define sample_run_Bridging_Header_h  
  
#import <paytabs-ios/paytabs_ios.h>  
  
#endif /* sample_run_Bridging_Header_h */  
|
```

8. Clean your workspace and delete derived data before building.

## 3 Integrating the SDK with your application

### 3.1 SDK Parameters

Element	Description	Format
XIB name	<u>Required</u> SDK initial .xib to invoke it.	String It's value should always be <i>PTFWInitialSetupView</i> .
Bundle	<u>Required</u> Resources bundle of SDK which contains images, xibs, local data files, etc.	NSBundle/Bundle See example code to check how it is passed in SDK.
View frame	<u>Required</u> CGRect (x, y, width, height) in which is sdk is intended to show.	CGRect E.g. <i>self.view.frame</i>
Amount	<u>Required</u> The amount of the transaction	Float Up to 3 Decimal places E.g.: 134.999
Customer title	<u>Required</u> Name of your customer.	String Up to 128 characters E.g.: <i>PayTabs iOS Store</i>



Tax Amount	<u>Required</u> The tax amount of the transaction	Float Up to 3 Decimal places E.g.: 134.999
Currency Code	<u>Required</u> Currency of the amount stated. 3 character ISO currency code	String E.g.: <i>USD for US dollar</i> <i>AED for Emirati Dirham</i> <i>BHD for Bahraini Dinar</i> <i>SAR for Saudi Riyal</i>
SDK Language	<u>Required</u> Use to set language of SDK.	String Either “ar” or “en”
Shipping Address	<u>Required</u> Shipping address of the customer. In case you don’t have physical goods, you can pass the same details as the billing details	String Up to 40 characters E.g.: <i>Flat 11 Building 222</i> <i>Block 333 Road 444 Manama</i> <i>Bahrain</i>
Shipping City	<u>Required</u> Shipping City of the customer	String Up to 13 characters E.g.: <i>Manama</i>
Shipping Country	<u>Required</u> Shipping country of the customer	String 3 character ISO country code E.g.: <i>BHR</i>
Shipping State	<u>Required</u> Shipping State of the customer 2 characters ISO state code for US and Canada	String 32 characters E.g.: <i>Manama</i> E.g.: if state is New York then the state field should be NY

Shipping ZIP code	<u>Required</u> Shipping postal code of the customer	String From 5 to 9 characters E.g.: 12345
Billing Address	<u>Required</u> Complete Address of the customer. Multiple address lines will be merged into one single line.	String Up to 40 characters E.g.: <i>Flat 11 Building 222 Block 333 Road 444 Manama Bahrain</i>
Billing City	<u>Required</u> Name of the billing city selected by customer	String Up to 13 characters E.g.: <i>Manama</i>
Billing Country	<u>Required</u> 3 character ISO Country of the customer	String 3 character ISO country code E.g.: <i>BHR for Bahrain ARE for UAE SAU for Saudi Arabia</i>
Billing State	<u>Required</u> Billing State (part of the address) entered by the customer 2 characters ISO state code for US and Canada	String Up to 32 characters E.g.: <i>Manama</i> E.g.: if state is New York, then the state field should be NY
Billing ZIP code	<u>Required</u> Billing Postal code provided by the customer. In case the country doesn't have a valid postal code, you can pass the country international dialing code (00973)	String From 5 to 9 characters E.g.: 12345
Order ID	<u>Required</u> The order id from your App to be able to map your order's to PayTabs transactions	String 32 characters E.g.: 4578

Phone Number	<u>Required</u> Phone Number of the Customer	String 32 characters E.g.: 00973312345678
Customer Email	<u>Required</u> Email of the Customer	String 32 characters E.g. customer_email@paytabs.com
Merchant Email	<u>Required</u> Merchant email that you use to sign up and/or login into PayTabs Merchant Dashboard	String Up to 128 characters E.g.: merchant@myapp.com
Secret Key	<u>Required</u> SECRET Key generated by merchant dashboard this can be found by logging in to your Merchant Dashboard > Mobile Payments > Secret Key	String Up to 128 characters E.g.: Mao8sdhasfyahosdap9sd
Timeout Seconds	<u>Required</u> Payment page will timeout after this time interval in seconds	Int E.g.: 200

## 3.2 Integration steps and guidelines

The following steps show what is needed to be able to integrate your application with the SDK,

1. Create instance variable for **PTFWInitialSetupViewController** in your UIViewController

```
// MARK: Instance Variables
var initialSetupViewController: PTFWInitialSetupViewController!
```

2. Implement events for initiating and closing SDK. (follow the sample app for the examples)

```
// MARK: Callbacks
private func configureCallbacks() -> Void {
    weak var weakSelf = self

    self.launcherView.didPressHitTestRunButtonCallback = {
        weakSelf?.handleHitTestRunTapEvent()
    }

    self.launcherView.didPressResponseViewCloseButtonCallback = {
        weakSelf?.handleCloseResponseViewTapEvent()
    }
}

// MARK: Event Methods
// MARK: Initiate SDK Event
private func handleHitTestRunTapEvent() {
    if (self.launcherView.amountTextField.text?.count == 0) {
        let alertController = UIAlertController(title: "Alert", message:
            "Amount is empty!", preferredStyle: UIAlertControllerStyle.alert)
        alertController.addAction(UIAlertAction(title: "Dismiss", style: UIAlertActionStyle.default,handler:
            nil))

        self.present(alertController, animated: true, completion: nil)
    } else {
        self.initiateSDK()

        self.view.addSubview(initialSetupViewController.view)
        self.addChildViewController(initialSetupViewController)

        initialSetupViewController.didMove(toParentViewController: self)
    }
}

// MARK: Close SDK Event
private func handleBackButtonTapEvent() {
    self.initialSetupViewController.willMove(toParentViewController: self)
    self.initialSetupViewController.view.removeFromSuperview()
    self.initialSetupViewController.removeFromParentViewController()
}

// MARK: Close Response window Event
private func handleCloseResponseViewTapEvent() {
    self.launcherView.responseView.isHidden = true
}
```

3. Initiate SDK on the relevant action e.g. it is being initiated on the tap of **hit test button** after some basic validations.

4. Call custom **init** static function of **PTFWInitialSetupViewController** with the values and by keeping the above mentioned notes in consideration.

```
// MARK: Objects
private func initiateSDK() {
    let bundle = Bundle(url: Bundle.main.url(forResource: "Resources", withExtension: "bundle")!)

    self.initialSetupViewController = PTFWInitialSetupViewController.init(
        nibName: "PTFWInitialSetupView",
        bundle: bundle,
        andWithViewFrame: self.view.frame,
        andWithAmount: 1.0,
        andWithCustomerTitle: "PayTabs",
        andWithCurrencyCode: "SAR",
        andWithTaxAmount: 0.0,
        andWithSDKLanguage: "en",
        andWithShippingAddress: "jeddah KSA",
        andWithShippingCity: "jeddah",
        andWithShippingCountry: "BHR",
        andWithShippingState: "123",
        andWithShippingZIPCode: "097",
        andWithBillingAddress: "Manama Bahrain",
        andWithBillingCity: "Manama",
        andWithBillingCountry: "BHR",
        andWithBillingState: "Manama",
        andWithBillingZIPCode: "097",
        andWithOrderID: "00987",
        andWithPhoneNumber: "0097335532915",
        andWithCustomerEmail: "humayun4206@gmail.com",
        andWithCustomerPassword: "xVoPaGMCaL",
        andIsTokenization: true,
        andIsExistingCustomer: false,
        andWithPayTabsToken: "s7oeTQxwuGpNGjFymzPSEWafapeIzBG1",
        andWithMerchantEmail: "your_email_address@domain.com",
        andWithMerchantSecretKey: "LWwm4jDZKOxg62YGVm3s1gJXegoRmfg8cNvg68gRXN84P6KLT0dV4wSjMha9xriJeN8lWgcT4kYUjrJG1xTgCq8Vcq6mXxEHlsH6o",
        andWithRequestTimeoutSeconds: 200,
        andWithAssigneeCode: "SDK")
}
```

### 3.3 SDK Response

Implement two callbacks, **didReceiveBackButtonCallback** and **didReceiveFinishTransactionCallback** to close the SDK and handle the SDK response respectively.

```
weak var weakSelf = self
self.initialSetupViewController.didReceiveBackButtonCallback = {
    weakSelf?.handleBackButtonTapEvent()
}

self.initialSetupViewController.didReceiveFinishTransactionCallback = {(responseCode, result, transactionID,
    tokenizedCustomerEmail, tokenizedCustomerPassword, token, transactionState) in
    self.launcherView.responseCodeLabel.text = "\(responseCode)"
    self.launcherView.resultLabel.text = "\(result)"
    self.launcherView.transactionIDLabel.text = "\(transactionID)"
    self.launcherView.customerEmailLabel.text = "\(tokenizedCustomerEmail)"
    self.launcherView.customerPasswordLabel.text = "\(tokenizedCustomerPassword)"
    self.launcherView.transactionStateLabel.text = "\(transactionState)"
    self.launcherView.tokenValueLabel.text = "\(token)"

    self.launcherView.responseView.isHidden = false

    weakSelf?.handleBackButtonTapEvent()
}
```

The callback, **didReceiveFinishTransactionCallback** contains seven values in the response after the transaction is processed:

Value	Variable	Description
Response Code	responseCode	Response code of the transaction.
Result	result	Result or description of the transaction.
Transaction ID	transactionID	Transaction identifier for tracking

		the transaction.
Tokenized Email	tokenizedCustomerEmail	Email address which has been tokenized if tokenization is enabled for future use.
Tokenized Password	tokenizedCustomerPassword	Password which has been tokenized if tokenization is enabled for future use.
Transaction State	transactionState	State of the transaction, either true or false.
Token	token	Token which is returned for future transactions.

The possible values for the response code variable are as follows:

Error Code	Description
100	Payment is completed Successfully.
481	<p>This transaction may be suspicious, your bank holds for further confirmation. Payment Provider has rejected this transaction due to suspicious activity; Your bank will reverse the dedicated amount to your card as per their policy.</p> <p><b>Important Note:</b> These transactions are rejected by our fraud monitoring tools; however the transactions are only authorized (Not settled). In case you verify the transaction is genuine you can contact us to force accept the transaction after checking if it is not fraud and we might ask verification documents like ID card and front side of credit card used showing only name and last 4 digits, otherwise the transaction amount will be released back to the cardholder according to the issuing bank policy. In case you are using IPN, it will post the response once the transaction is force Accepted from our side</p>
Any other response code	Any other response means that the transaction has been rejected and the rejection reason is the description of the corresponding response code
**	The Complete list of response codes and their description can be found in Merchant Dashboard > Help > Reason Codes

### 3.4 Language Configuration

In **sdklanguage** you can set the language preference. For Arabic, it will be “ar” and for English, it will be “en”. By default, it will be “en”.

Currently PayTabs\_SDK has only two languages

- English (“en”)
- Arabic (“ar”)

### 3.5 Important Notes

- 1 There shouldn't be any special characters in address related parameters i.e. characters should be from the set of A-Z, a-z, 1-9 or a space.
- 2 Provided **country codes** should be from the list of valid **3 ISO Country codes**. Valid country codes are against the **alpha-3** key within the dictionaries for countries in *countries.json* which you can find in **Resources bundle**.
- 3 Provided **currency code** should be **3 character code**. Valid currency codes are against the **code** key within the dictionaries for currencies in *currencies.json* which you can find in **Resources bundle**.
- 4 Make sure you are not passing any empty string against any value.
- 5 **Merchant and customer emails** should follow proper email format.
- 6 Optional parameter, **Request timeout** should be more than 15 seconds.
- 7 **SDK Language** should be either “en” or “ar”.
- 8 **Phone Number** should start with either “+” or “00” and should not have anything except digits.
- 9 **Amount** should be in proper decimal number format and should be greater than 0.

## 4 Verify Transaction

This method verifies the result of the payment. When a payment is processed, you will receive transaction id in the response. When the customer completes a payment and is redirected back to your app, the transaction\_id is used to verify the status of the payment whether it is a successful transaction or a failed transaction.

<b>PATH</b>	https://www.paytabs.com/apiv2/verify_payment_transaction
<b>REQUEST METHOD</b>	POST
<b>PRODUCTION</b>	LIVE

### 4.1 Request

Element	Description	Format
merchant_email	<u>Required</u> Merchant email that you use to sign up and/or login into PayTabs Merchant Dashboard	String Up to 128 characters E.g.: foof@foo.com

secret_key	<u>Required</u> SECRET Key generated by merchant dashboard	String Up to 128 characters E.g.: <i>Mao8sdhasfyahosdap9sd</i>
transaction_id	<u>Required</u> This is the transaction_id that is returned in the response of the payment.	String Up to 12 characters E.g.: <i>130017</i>
order_id	<u>Optional</u> This is the order id from your system, you can use that instead of transaction_id parameter	String 32 characters E.g.: <i>4578</i>

## 4.2 Response

<b>Element</b>	<b>Description</b>	<b>Format</b>
result	The string representation for result, whether transaction has succeeded or failed and the reason why your transaction failed.	String E.g.: <i>Payment is completed.</i>
response_code	The response code / error code of the API call. Complete list of response codes can be found in Merchant Dashboard > Help > Reason Codes	String Up to 4 characters E.g.: <i>4002</i>
pt_invoice_id	The invoice/PayPage ID of the transaction, this is available only for PayPage and API integration it will be null in case of mobile SDKS transactions	String Up to 12 characters E.g.: <i>null</i>
amount	The transaction amount.	Decimal Up to 3 Decimal places E.g.: <i>123.399</i>
currency	The currency of the transaction amount.	String 3 character ISO country code E.g.: <i>BHR</i>
transaction_id	The ID of the latest transaction that occurred for this invoice	String Up to 12 characters E.g.: <i>90094</i>
order_id	This is the order id from your system.	String 32 characters E.g.: <i>4578</i>
card_last_four_digits	This is the last four digits of the credit card used in the transaction	String 32 characters



		E.g.: 1111
--	--	------------

### 4.3 Response Codes

Error Code	Description
4001	Missing parameters
4002	Invalid Credentials
4003	There are no transactions available.
0404	You don't have permissions
100	Payment is completed Successfully.
481	<p>This transaction may be suspicious, your bank holds for further confirmation. Payment Provider has rejected this transaction due to suspicious activity; Your bank will reverse the dedicated amount to your card as per their policy.</p> <p><b>Important Note:</b> These transactions are rejected by our fraud monitoring tools; however the transactions are only authorized (Not settled). In case you verify the transaction is genuine you can contact us to force accept the transaction after checking if it is not fraud and we might ask verification documents like ID card and front side of credit card used showing only name and last 4 digits, otherwise the transaction amount will be released back to the cardholder according to the issuing bank policy. In case you are using IPN, it will post the response once the transaction is force Accepted from our side</p>
**	The Complete list of response codes can be found in Merchant Dashboard > Help > Reason Codes

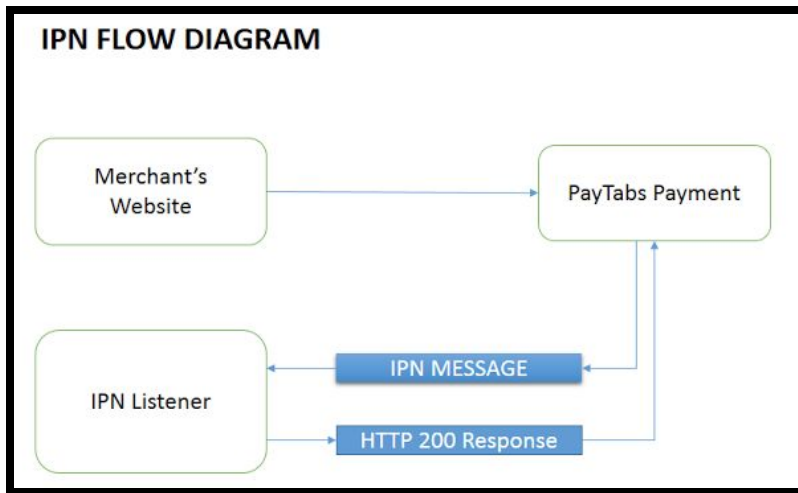
## 5 PayTabs IPN (Instant Payment Notification)

### 5.1 How IPN Works

When a customer makes a payment on the mobile SDK, the transaction is processed by PayTabs and a confirmation is sent back to the customer on the merchant's application. Additionally, a notification email will be sent to the merchant.

This scenario, leave no scope for any related ERP or E-commerce or Accounting Systems to be updated. This is where IPN comes in.

As soon as a transaction is created or altered, PayTabs IPN will instantly send a notification to the merchant's specified IPN Listener URL, which can have any code that will assist in updating ERP or dependent systems. Alternatively, the merchant may wish to save this information in his local database.



## 5.2 Set up your IPN Listener

1. Create an IPN Listener that receives and processes the IPN messages sent by PayTabs.
2. Log into your PayTabs Merchant Account > Go to “My Profile” and click on “Edit”.
3. Enter the URL of your IPN Listener and click on “Submit”. IPN will not work with a local host IPN Listener URL. Testing the IPN Message Service is only possible on a publicly hosted URL.

The screenshot shows the 'Edit' profile page in the PayTabs merchant account. The 'Paypage Currency' is set to 'USD'. The 'IPN Listener URL' field is highlighted with a red rectangle. Below it, there is a field for 'IPN Listener URL for Transactions Reports'. At the bottom, there is a 'Payment Timeout (Seconds)' section with a checked 'No Timeout' option. The 'Submit' button is highlighted in blue.

Now your IPN Service is ready to go!

## 5.3 IPN Events

Your listener will be triggered on the following events:

1. Successful Transaction
2. Rejected Transaction
3. Transaction, Force Accepted by PayTabs
4. Refunded Transaction

## 5.4 Contents / Data of the IPN

The Instant Payment Notification will return the following information depending on the type of event that has occurred.

<i>Element</i>	<i>Description</i>	<i>Format</i>
transaction_id	PayTabs Transaction ID	String E.g.: 12345
reference_id	Merchant's Website order ID, this the same as reference_no value posted in Create PayPage call	String Up to 12 characters E.g.: 90094
invoice_id	The PayPage id of the successful payment that you wish to refund. This is the p_id that is returned in the create PayPage response.	String Up to 12 characters E.g.: 90094
response_code	The response code / error code of the API call. A complete list of response codes can be found in Merchant Dashboard > Help > Reason Codes	String Up to 4 characters E.g.: 4002
customer_name	Name of the Customer	String 32 characters E.g.: John
email	Email of the customer	String 32 characters E.g.: customer@domain.com
phone_num	Phone Number of the Customer	String Minimum 6 numbers up to 32 numbers E.g.: 3312345678
amount	The transaction amount.	Decimal Up to 3 Decimal places E.g.: 123.399
currency	Currency of the amount stated. 3 character ISO currency code	String 3 characters E.g.: BHD
shipping_address	Shipping address of the customer	String 40 characters E.g.: Flat abc road 123
shipping_city	Shipping City of the customer	String 13 characters E.g.: Manama
shipping_state	Shipping State of the customer	String 32 characters E.g.: Manama 2 characters for US & Canada

		E.g.: if state is New York then the state field should be NY
shipping_country	Shipping country of the customer.	String 3 character ISO country code E.g.: <i>BHR</i>
shipping_postalcode	Shipping postal code of the customer	String From 5 to 9 characters E.g.: <i>403129</i>
order_id	The order id from your system to be able to map your order's to PayTabs transactions	String 32 characters E.g.: <i>4578</i>

## 5.5 Response Codes

Error Code	Description
5000	Payment has been rejected
5001	Payment has been accepted successfully
5002	Payment has been forcefully accepted
5003	Payment has been refunded

### Important Notes

1. When merchants use Verify transaction API, the API will send a response for successful or rejected payment. The IPN feature will also give the same response. The merchant should be cautious and accept both notifications as one transaction and not separate transactions. Moreover, **ONLY** the IPN will post updated information about force accepted transactions.

## 5.6 Sample IPN Response

### 5.6.1 IPN sample response for successful SDK transactions

Array

(

```
[transaction_id] => 263079
[order_id] => 1234567
[response_code] => 100
[customer_name] => John Doe
[customer_email] => John@gmail.com
[transaction_amount] => 1.00
```

```

[transaction_currency] => USD
[customer_phone] => 009733
[last_4_digits] => 1111
[first_4_digits] => 4111
[card_brand] => Visa
[datetime] => 06-04-2017 04:21:19 PM
[shipping_address] => Flat 1,Building 123, Road 2345
[shipping_city] => Juffair
[shipping_country] => BHR
[shipping_state] => Manama
[shipping_postalcode] => 00973
[amount] => 1.00
[currency] => USD
[detail] => Transaction has been Successful
)

```

### 5.6.2 IPN sample response for rejected SDK transactions

Array

```

(
    [transaction_id] => 263079
    [order_id] => 1234567
    [response_code] => 202
    [customer_name] => John Doe
    [customer_email] => John@gmail.com
    [transaction_amount] => 1.00
    [transaction_currency] => USD
    [customer_phone] => 009733
    [last_4_digits] => 1111
    [first_4_digits] => 4111
    [card_brand] => Visa
    [datetime] => 06-04-2017 04:21:19 PM
    [shipping_address] => Flat 1,Building 123, Road 2345
    [shipping_city] => Juffair
    [shipping_country] => BHR
    [shipping_state] => Manama
    [shipping_postalcode] => 00973
    [amount] => 1.00
    [currency] => USD
    [detail] => An expired card has been used in this transaction
)

```

### 5.6.3 IPN sample response for rejected SDK transactions by fraud monitoring

Array

```

(
    [transaction_id] => 263079
    [order_id] => 1234567
    [response_code] => 481
    [customer_name] => John Doe

```

```

[customer_email] => John@gmail.com
[transaction_amount] => 1.00
[transaction_currency] => USD
[customer_phone] => 009733
[last_4_digits] => 1111
[first_4_digits] => 4111
[card_brand] => Visa
[datetime] => 06-04-2017 04:21:19 PM
[shipping_address] => Flat 1, Building 123, Road 2345
[shipping_city] => Juffair
[shipping_country] => BHR
[shipping_state] => Manama
[shipping_postalcode] => 00973
[amount] => 1.00
[currency] => USD
[detail] => This transaction may be suspicious, your bank holds for further
confirmation. Payment Provider has rejected this transaction due to suspicious
activity; your bank will reverse the dedicated amount to your card as per their
policy.
)

```

#### 5.6.4 IPN sample response for force accepted SDK transactions

Array

```

(
    [transaction_id] => 263076
    [order_id] => 25
    [response_code] => 5002
    [customer_name] => John Doe
    [customer_email] => john@email.com
    [transaction_amount] => 1.00
    [transaction_currency] => USD
    [customer_phone] => 973 123456
    [last_4_digits] => 1111
    [first_4_digits] => 4111
    [card_brand] => Visa
    [secure_sign] => 7ec0ac0890acbe33139efe989d32e97e061f93f4
    [force_accept_datetime] => 06-04-2017 12:01:40 PM
)

```

#### 5.6.5 IPN sample response for refunded SDK transactions

Array

```

(
    [transaction_id] => 263076
    [order_id] => 25
    [response_code] => 5003

```

```

[customer_name] => John Doe
[customer_email] => john@email.com
[transaction_amount] => 100.00
[transaction_currency] => USD
[customer_phone] => 973 123456
[last_4_digits] => 1111
[first_4_digits] => 4111
[card_brand] => Visa
[secure_sign] => 7ec0ac0890acbe33139efe989d32e97e061f93f4
[refund_req_amount] => 50
)

```

## 6 Refund Transactions

This method enables you to submit the refund requests using the below API function. Refund requests will be processed in around 24 hours and if you are using IPN service the IPN will post a response once the refund request is approved or rejected. Partial refunds are available. You can submit a single refund request for the whole amount of Multiple Partial refunds for the same transactions. However, if your refund for a certain transaction is still pending for approval, you can't submit a second refund request for the same transaction. All refund requests submitted using the API is also available at the Merchant Dashboard.

Note: This feature is also available from Merchant Dashboard > Bank Account > Refunds

Note: This API can be used by Active accounts only; it's not available for Demo merchants

<b>PATH</b>	<a href="https://www.paytabs.com/apiv2/refund_process">https://www.paytabs.com/apiv2/refund_process</a>
<b>REQUEST METHOD</b>	POST
<b>PRODUCTION</b>	LIVE

### 6.1 Request

<i>Element</i>	<i>Description</i>	<i>Format</i>
merchant_email	<u>Required</u> Merchant email that you use to sign up and/or login into PayTabs Merchant Dashboard	String Up to 128 characters E.g.: <i>foof@foo.com</i>
secret_key	<u>Required</u> SECRET Key can be found at merchant dashboard	String Up to 128 characters E.g.: <i>Mao8sdhasfyahosdap9sd</i>
transaction_id	<u>Optional</u> The transaction id of the successful payment that you wish to refund	String Up to 12 characters E.g.: <i>130017</i>

	This is the pt_transaction_id that is returned in the SDK response of the payment.	
order_id	<u>Optional</u> The order id of the payment from your system You can either use this or use the transaction_id	String Up to 32 characters E.g.: 4578
refund_amount	<u>Required</u> The amount you wish to refund, it can either be partial amount or full amount	Up to 3 Decimal places E.g.: 134.999
refund_reason	<u>Required</u> The reason for refunding the transaction	String Up to 256 Characters

## 6.2 Response

<b>Element</b>	<b>Description</b>	<b>Format</b>
result	The string representation for result, whether transaction has succeeded or failed and the reason why your transaction failed.	String E.g.: <i>Payment is completed.</i>
response_code	The response code / error code of the API call. Complete list of response codes can be found in Merchant Dashboard > Help > Reason Codes	String Up to 4 characters E.g.: 4002

## 6.3 Response Codes

<b>Error Code</b>	<b>Description</b>
<b>4001</b>	Missing parameters
<b>4002</b>	Invalid Credentials
<b>810</b>	You already requested Refund for this Transaction ID
<b>811</b>	Amount is above or below the invoice and also the minimum balance



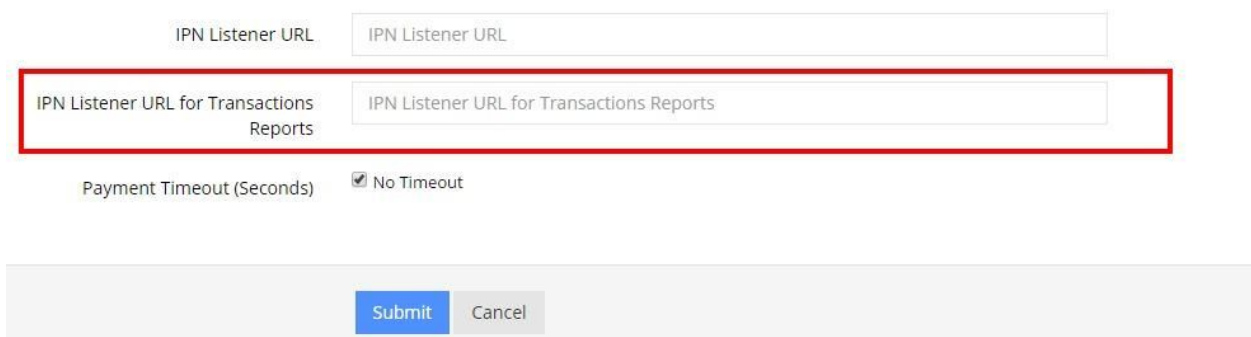
<b>812</b>	Refund request is sent to Operation for Approval. You can track the Status
<b>813</b>	You are not authorized to view this transaction

## 7 Transactions Reports

This API call will post all transactions that have taken place within the specified time interval to a URL.

Before calling this API, you will need to set the listener URL for the reports. Login to your merchant dashboard and edit your profile. Set up your IPN Listener

1. Create an IPN Listener that receives and processes the IPN messages sent by PayTabs.
2. Log into your PayTabs Merchant Account > Go to “My Profile” and click on “Edit”.
3. Enter the URL of your IPN Listener In the ‘IPN Listener for Transactions Reports’ and click on “Submit”. IPN will not work with a local host IPN Listener URL. Testing the IPN Message Service is only possible on a publicly hosted URL.



IPN Listener URL

IPN Listener URL

IPN Listener URL for Transactions Reports

IPN Listener URL for Transactions Reports

Payment Timeout (Seconds) ☒ No Timeout

Submit Cancel

Once you call the transaction\_reports API, all the transactions will be posted to the listener URL and the response will contain the number of transactions that will be posted.

<b>PATH</b>	<a href="https://www.paytabs.com/apiv2/transaction_reports">https://www.paytabs.com/apiv2/transaction_reports</a>
<b>REQUEST METHOD</b>	POST
<b>PRODUCTION</b>	LIVE

### 7.1 Request

Element	Description	Format
merchant_email	<u>Required</u> Merchant email id	String E.g.: <i>foo@foo.com</i>
secret_key	<u>Required</u> Secret Key is generated from merchant Dashboard.	String E.g.: <i>Mao8sdhasfyahosdap9sd</i>
startdate	<u>Required</u> Start Date Format should be dd-mm-yy.	String E.g.: <i>13-01-2015</i>

enddate	<u>Required</u> End Date Format should be dd-mm-yy.	String E.g.: 13-01-2015
---------	--	----------------------------

## 7.2 Response

<b>Element</b>	<b>Description</b>	<b>Format</b>
Response_code	Response code of the API call	String E.g.: 4001, 4090, 4091
transaction_count	Total number of transactions.	String E.g.: 23

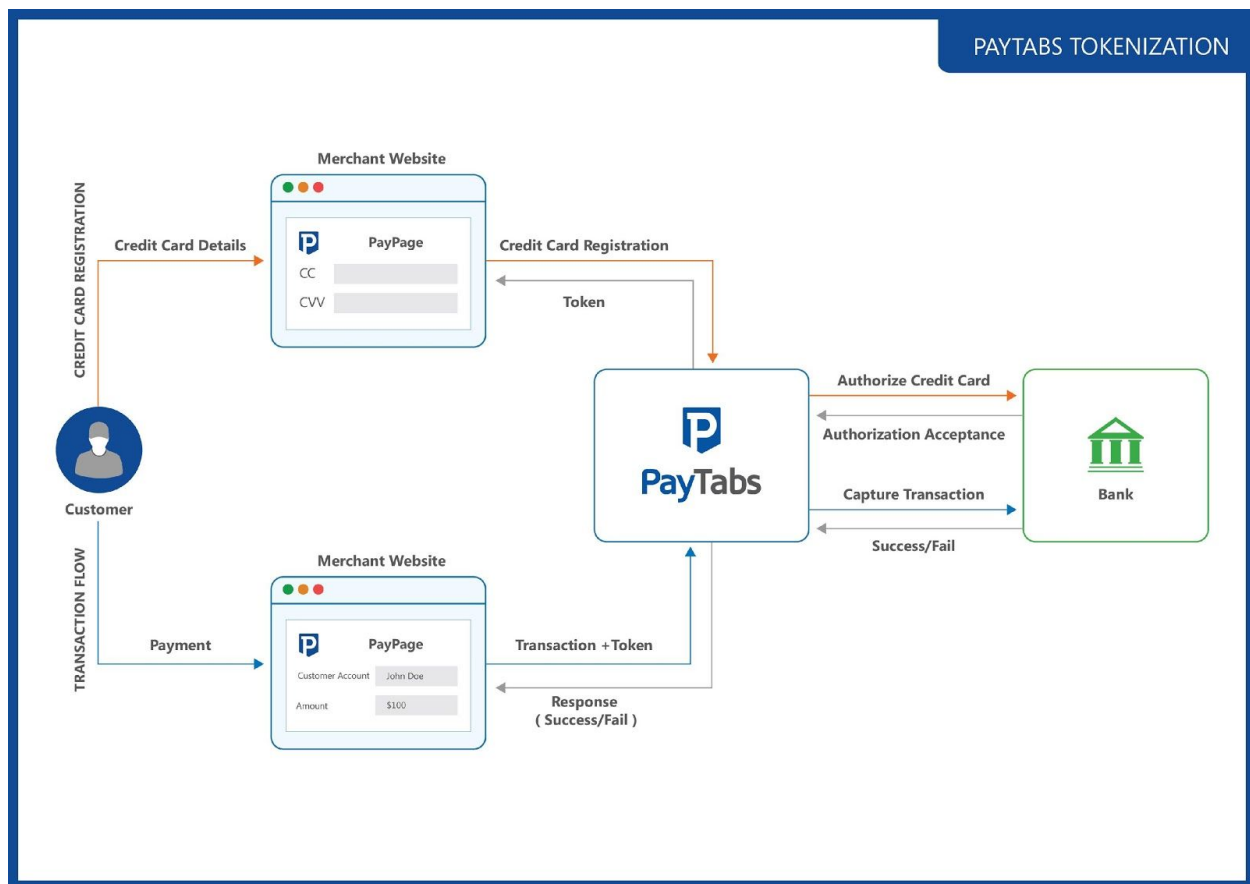
## 7.3 Error Codes

<b>Error Code</b>	<b>Description</b>
4001	Missing parameters
4002	Invalid Credentials
4090	Data Found
4091	No Data

## 8 Tokenization in SDK

The Tokenization feature gives you the option to create tokenization profiles to save credit card information for returning customers to your website.

**Please Note: Tokenization feature is not available for merchants registered or operating in the Kingdom of Saudi Arabia.**



In order to use tokenization in the SDK, you will need to pass additional parameters in the method. This will result in the creation of a normal Payment request, where the customer has to perform and authorize the first payment and receive the token for the credit card details that he has entered.

The following parameters have to be added to the creation of payment request in order to create a token for the customer.

Element	Description	Format
Is Tokenization	<u>Optional</u> Set this value as true if you wish to create tokenization profile for customer or use an existing token	String E.g.: true , false

Is Existing Customer	<u>Optional</u> False: If you want to create a new token	String E.g.: true, false
PayTabs Token	<u>Optional</u> This pt_token is received in the API post response after completing the payment; it will be redirected to return_url. While returning back to that URL, it will send a POST request to that page	String E.g.: R7ANsPK1q91fv5Q0bmQ3
Customer Email	<u>Optional</u> The customer email linked to the tokenization profile used, by default when the profile is created at the first successful payment, it will use customer_email value sent in the API to link it to the token	String E.g.: <i>customer@domain.com</i>
Customer Password	<u>Optional</u> This pt_customer_password is received in the API post response after completing the payment; it will be redirected to return_url. While returning back to that URL, it will send a POST request to that page.	String E.g.: 1q91fv5Q0b

## 9 Test Credit Card Numbers

To test your payment process, you can typically use any valid credit card number or you can use the ones listed below.

Please do not use test cards in a live environment. This will cause your transactions to be rejected.

Card Number	Description	CVV	Expiry Date
4111111111111111	Visa Card	1234	01/18

5200000000000114	MasterCard	1234	01/18
------------------	------------	------	-------

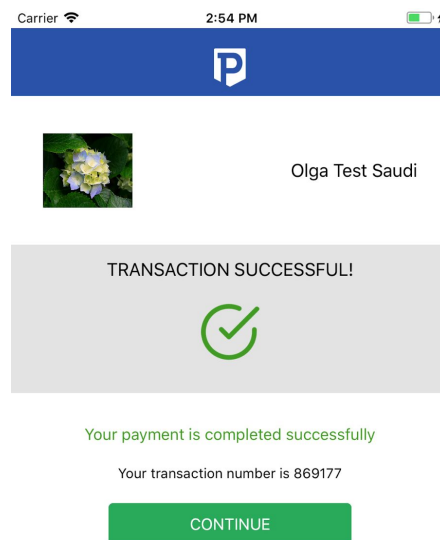
## 10 Screenshots of the application

### 10.1 PayTabs Mobile Payment Page

- The customer will be redirected to PayTabs payment page to enter the card details which will show as follows.
- The payment page will display the Logo you submitted at your PayTabs Merchant Account on the top left of the page and the Store Name submitted at your PayTabs Merchant Account on the top right.
- The required card information will be:
  - Card Number
  - Card Holder Name
  - Expiration Date (mm/yy)
  - CVV (Card Verification Code)

### 10.2 Successful Payment Result Page

- Once the customer submits the card information to process the payment, if the payment is successful, the customer will be redirected to the Successful result page that will return the successful transaction ID to the customer.



## 11 Frequently Asked Questions

### Q1. PayTabs iOS SDK is not working on actual iOS devices lesser than iPhone 5S!

Our SDK supports armv7, armv7s and arm 64.

### Q2. How do I get the response code from the iOS SDK?

To view transaction\_id and response codes please add the following code in your application.

```
NSLog(@"transaction Id %@", [[NSUserDefaults standardUserDefaults]
objectForKey:@"pt_transaction_id"]);
```

```
NSLog(@"Response Code %@", [[NSUserDefaults standardUserDefaults]
objectForKey:@"pt_response_code"]);
NSLog(@"Description %@", [[NSUserDefaults standardUserDefaults] objectForKey:@"pt_description"]);
```

The possible values for the response code variable are as follows:

Error Code	Description
100	Payment is completed Successfully.
481	<p>This transaction may be suspicious, your bank holds for further confirmation. Payment Provider has rejected this transaction due to suspicious activity; Your bank will reverse the dedicated amount to your card as per their policy.</p> <p><b>Important Note:</b> These transactions are rejected by our fraud monitoring tools; however the transactions are only authorized (Not settled). In case you verify the transaction is genuine you can contact us to force accept the transaction after checking if it is not fraud and we might ask verification documents like ID card and front side of credit card used showing only name and last 4 digits, otherwise the transaction amount will be released back to the cardholder according to the issuing bank policy. In case you are using IPN, it will post the response once the transaction is force Accepted from our side</p>
Any other response code	Any other response means that the transaction has been rejected and the rejection reason is the description of the corresponding response code
**	The Complete list of response codes and their description can be found in Merchant Dashboard > Help > Reason Codes

### Q3. Is the IOS SDK compatible with Swift?

Yes, it is compatible with both languages i.e. Objective-C and Swift. But, in order to use within Swift based codebase, you will need to set up Obj-C bridging header.

### Q4. Can I use the iOS SDK in my hybrid framework like Cordova, Sencha, Titanium, Ionic?

Unfortunately, our iOS SDK does not support hybrid frameworks.

### Q5. Does your iOS SDK work on simulators?

Yes, you can run iOS SDK on simulators.

### Q6. How can I get the product names from Verify Payment API & IPN response from SDK?

The product names can be stored in your database against your own invoice id/reference id/order Id. Once the payment is made, we will send the same id back to your server in the verify\_payment / IPN response. You can then compare the IDs and retrieve the product names from your own database.

### Q7. May I know from which iOS version it will work?

The SDK is compatible till latest iOS 11.

### Q8. Why is the design of the 3DS page not centralized on the screen?

This screen is coming directly from the bank. So the control over this page is very limited. We have added a scroll bar so that users will be able to scroll the page.

### Q9. During payment, the app waits on a page for a long time before proceeding to the next step. Can we add a loading image here?

The loader has been added to the LIVE library only due to some restrictions in the test environment. So once you are LIVE you will be able to see the loader.

### Q10. While testing on test account, some credit card transactions seem to have an issue... the loader keeps on loading. How can we fix this?

On test environment, we have some limitations on amounts such as 6, 60, 66 etc. Kindly attempt different amounts that do not contain the digit "6".

### Q11. How can we test refund transactions in both Android and iOS. Is there any way to test with one test card?

Refund requests cannot be submitted for demo transactions. You will need to refund request for a LIVE transaction but kindly limit the same to 1 or 2 transactions.





PayTabs iOS SDK – Version 3.0

<https://www.paytabs.com>