

iOS Native SDKs Kit Guide

For simpler in-app payments.

This document is meant for PG integration on iOS Platform.

Copyright (c) 2014 Citrus Pay. All rights reserved.

Accepting payments in your app involves 3 steps, which we'll cover in this guide:

- Installation
- Get the Bill generator
- Using Payment Options

What type of payments can I make with Citrus?

You can pay using cards, card tokens (credit/debit) and netbanking.

Where do I get started?

The kit provided to you contains an Example and CitrusDevKit/. follow the installation screens below,

Installation

Requirements

- Xcode 6 and iOS SDK 8
- iOS 6.0+ target deployment
- armv7, armv7s, and arm64 devices, and the simulator
- iPhone and iPad of all sizes and resolutions

Using git

Get the latest source code from github.com:

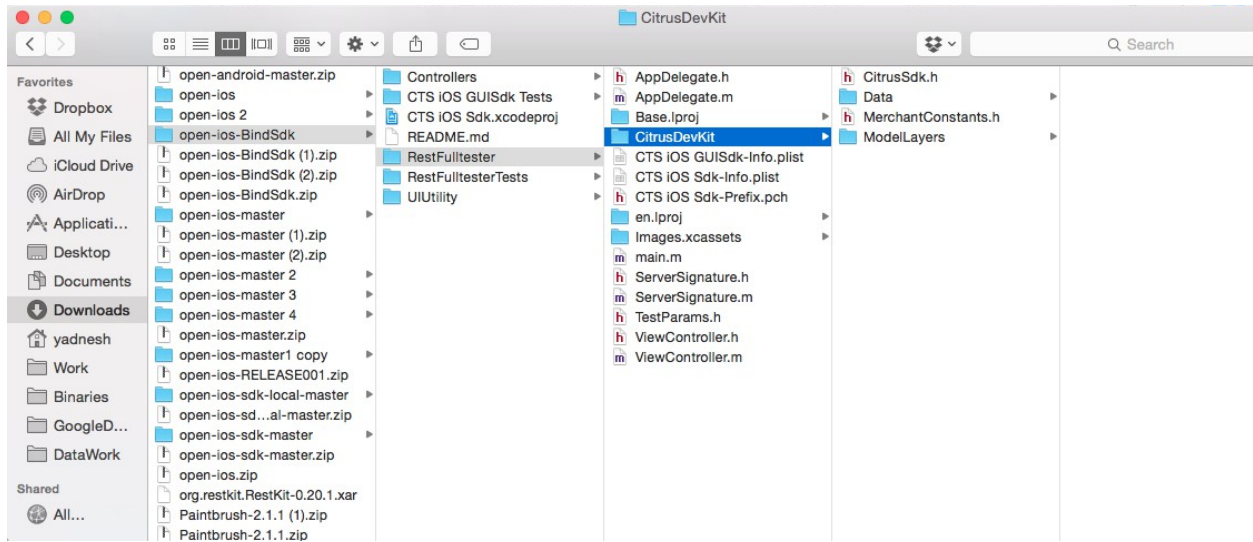
```
$ git clone -b BindSdk https://github.com/citruspay/open-ios.git
```

open the zip file, having SDK folder "CitrusDevKit" inside.

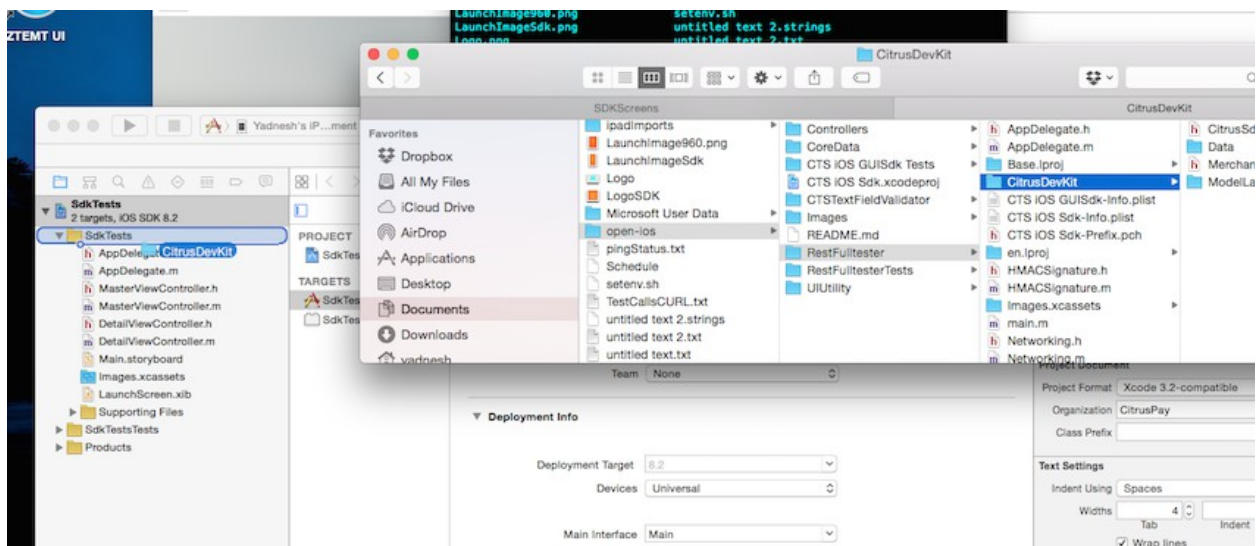
SimpleViewController.m contains sample code for SDK integration

This document is meant for PG integration on iOS Platform.

Copyright (c) 2014 Citrus Pay. All rights reserved.



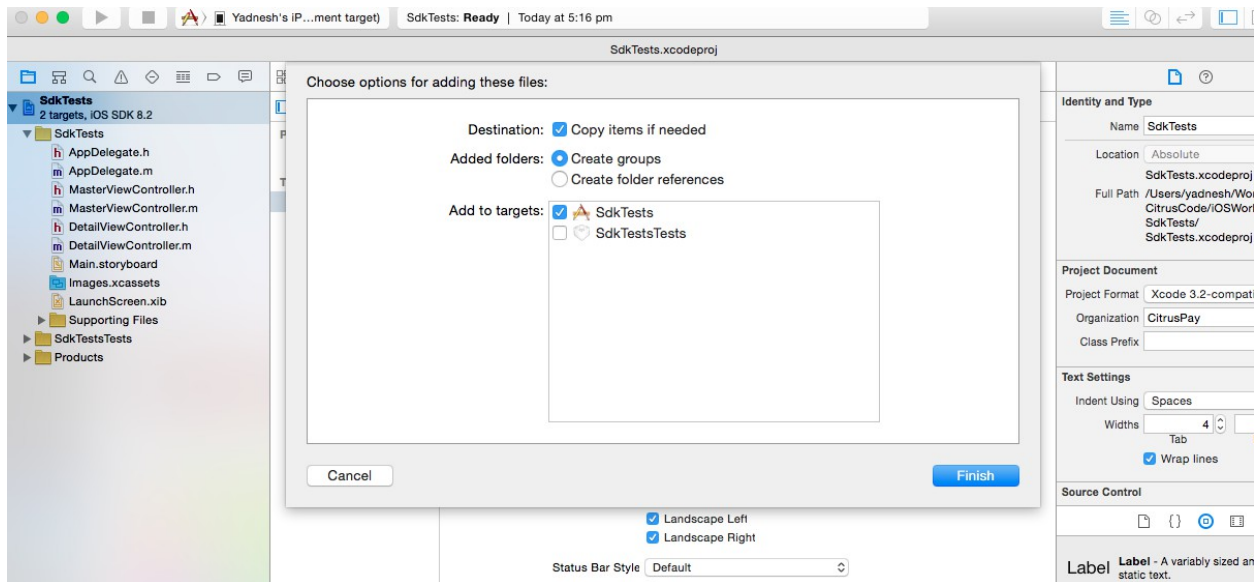
To install it in your project just drag drop CitrusDevKit to you project.



Add it as groups like this

This document is meant for PG integration on iOS Platform.

Copyright (c) 2014 Citrus Pay. All rights reserved.



```
#import "CitrusDevKit/CitrusSdk.h"
```

You are now done with the installations.

What do I need to make payment?

- 1) You need an access key and a secret key found inside the admin panel.
- 2) You need a bill generator hosted on your server.
- 3) You need a return url - where Citrus posts you the response of transaction.

How do I put the keys?

You have to put your keys and URLs in MerchantConstants.h file. Request to Citrus support for your keys - at [tech.support\[at\]citruspay.com](mailto:tech.support[at]citruspay.com).

//KEYS

```
#define VanityUrl @"nativeSDK"
#define SignInId @"oaut-test"
#define SignInSecretKey @"01248"
#define SubscriptionId @"citrus-native-mobile-subscription"
#define SubscriptionSecretKey @"3e2288d3a1a3f59ef6f93373884d2ca1"
```

//URLS

```
#define BillUrl @"http://YourServer/Bill"
```

This document is meant for PG integration on iOS Platform.

Copyright (c) 2014 Citrus Pay. All rights reserved.

What's a bill generator?

Citrus needs to be sure that the payment request has been originated by its merchant. Bill generator - as it is **hosted with merchant server** - makes sure that the bill is generated by merchant. That's why keeping your secret key secret, is extremely important.

You have to add your BillUrl in MerchantConstants.

Expected JSON from your server is in following format,

```
{
  "merchantTxnId": "141715943163769",
  "amount": {
    "value": "100",
    "currency": "INR"
  },
  "requestSignature": "9e50de5ccc61f30245db1f6f6f3fb55b76816ccc",
  "merchantAccessKey": "F2VZD1HBS2VVXJPMWO77",
  "returnUrl": "http://103.13.97.20/citrus/index.php"
}
```

Your app makes a call to this bill generator. The code for this call, **getBillFromServer**, is kept out of SDK incase you need to modify it can be found in Sample app's **SimpleViewController.m**.

```
/*
You can modify this according to your needs.
This is sample implementation.
*/
+ (CTSBill*)getBillFromServer{
    // Configure your request here.
    NSMutableURLRequest* urlReq = [[NSMutableURLRequest alloc] initWithURL:
                                     [NSURL URLWithString:BillUrl]];
    [urlReq setHTTPMethod:@"POST"];
    NSError* error = nil;
    NSData* signatureData = [NSURLConnection sendSynchronousRequest:urlReq
                                     returningResponse:nil
                                     error:&error];
    NSString* billJson = [[NSString alloc] initWithData:signatureData
                                     encoding:NSUTF8StringEncoding];
    JSONModelError *jsonError;
```

This document is meant for PG integration on iOS Platform.

Copyright (c) 2014 Citrus Pay. All rights reserved.

```

CTSBill* sampleBill = [[CTSBill alloc] initWithString:billJson
                        error:&jsonError];
NSLog(@"signature %@", sampleBill);
return sampleBill;
}

```

How do I make a payment?

- a. Get the bill from your server
- b. Collect user details
- c. Call the charge API.

Card can be initialized with full card details or with a token and cvv. Please note that we do not store cvv. It has to be collected from the user in the runtime.

Full card details:

You need to pass card number, exp month, exp year, cvv, card owner's name, and card type - credit or debit.

Card token details:

You need to send the card token along with the cvv.

// Card payment debit/credit

```

-(IBAction)cardPayment:(id)sender{
    CTSPaymentDetailUpdate *creditCardInfo = [[CTSPaymentDetailUpdate alloc] init];
    // Update card for card payment.
    CTSElectronicCardUpdate *creditCard = [[CTSElectronicCardUpdate alloc] initWithCreditCard];
    creditCard.number = TEST_CREDIT_CARD_NUMBER;
    creditCard.expiryDate = TEST_CREDIT_CARD_EXPIRY_DATE;
    creditCard.scheme = TEST_CREDIT_CARD_SCHEME;
    creditCard.ownerName = TEST_CREDIT_CARD_OWNER_NAME;
    creditCard.name = TEST_CREDIT_CARD_BANK_NAME;
    creditCard.cvv = TEST_CREDIT_CARD_CVV;
    [creditCardInfo addCard:creditCard];
}

```

// Get your bill here.

```

CTSBill *bill = [SimpleStartViewController getBillFromServer];

```

// Configure your request here.

This document is meant for PG integration on iOS Platform.

Copyright (c) 2014 Citrus Pay. All rights reserved.

```

    [paymentLayer requestChargePayment:creditCardInfo withContact:contactInfo
withAddress:addressInfo bill:bill withCompletionHandler:^(CTSPaymentTransactionRes
*paymentInfo, NSError *error) {
    [self handlePaymentResponse:paymentInfo error:error];
    }];
}

```

Netbanking:

You need to send the cid number of the bank. A detailed list of Banks and their cids is presented at the end of the document. `TEST_NETBAK_CODE` is macro to the cid.

After the charge API is called, you would get a redirect url in the callback passed. Open this redirect url in a `UIWebView` - it would take you to either a 3D secure or a netbanking page.

// Netbanking

```

-(IBAction)netbankingPayment:(id)sender{
    CTSPaymentDetailUpdate *paymentInfo = [[CTSPaymentDetailUpdate alloc] init];
    // Update bank details for net banking payment.
    CTSNetBankingUpdate* netBank = [[CTSNetBankingUpdate alloc] init];
    netBank.code = TEST_NETBAK_CODE;
    [paymentInfo addNetBanking:netBank];

    // Get your bill here.
    CTSBill *bill = [SimpleStartViewController getBillFromServer];

    // Configure your request here.
    [paymentLayer requestChargePayment:paymentInfo withContact:contactInfo
withAddress:addressInfo bill:bill withCompletionHandler:^(CTSPaymentTransactionRes
*paymentInfo, NSError *error) {
    [self handlePaymentResponse:paymentInfo error:error];
    }];
}

```

What's bind user?

Bind user creates the Citrus account of the user and allows you to save cards on user's behalf.

To save a card or to fetch wallet - you need to bind the user.

This document is meant for PG integration on iOS Platform.

Copyright (c) 2014 Citrus Pay. All rights reserved.

```

// Bind to the User.
-(IBAction)bindUser:(id)sender{
    // Configure your request here.
    [authLayer requestBindUsername:TEST_EMAIL mobile:TEST_MOBILE
    completionHandler:^(NSString *userName, NSError *error) {
        NSLog(@"error.code %ld ", (long)error.code);

        if(error == nil){
            // Your code to handle success.
            [UIUtility toastMessageOnScreen:[NSString stringWithFormat:@" %@ is now
bound",userName]];
        }
        else {
            // Your code to handle error.
            [UIUtility toastMessageOnScreen:[NSString stringWithFormat:@" couldn't bind
%@\\nerror: %@",userName,[error description]]];
        }
    }];
}

```

How do I save the card?

- a. Bind the user - (please refer above bind user section)
- b. Save the card - (We do not save the cvv)

```

// Save the cards.
-(IBAction)saveCards:(id)sender{
    CTSPaymentDetailUpdate *paymentInfo = [[CTSPaymentDetailUpdate alloc] init];
    // Credit card info for card payment type.
    CTSElectronicCardUpdate *creditCard = [[CTSElectronicCardUpdate alloc] initWithCreditCard];
    creditCard.number = TEST_CREDIT_CARD_NUMBER;
    creditCard.expiryDate = TEST_CREDIT_CARD_EXPIRY_DATE;
    creditCard.scheme = TEST_CREDIT_CARD_SCHEME;
    creditCard.ownerName = TEST_CREDIT_CARD_OWNER_NAME;
    creditCard.name = TEST_CREDIT_CARD_BANK_NAME;
    [paymentInfo addCard:creditCard];
    paymentInfo.defaultOption = TEST_CREDIT_CARD_BANK_NAME;
    [paymentInfo addCard:creditCard];
}

```

This document is meant for PG integration on iOS Platform.

Copyright (c) 2014 Citrus Pay. All rights reserved.


```

// Configure your request here.
[profileLayer updatePaymentInformation:paymentInfo withCompletionHandler:^(NSError
*error) {
    if(error == nil){
        // Your code to handle success.
        [UIUtility toastMessageOnScreen:@" successfully card saved "];
    }
    else {
        // Your code to handle error.
        [UIUtility toastMessageOnScreen:[NSString stringWithFormat:@" couldn't save card\n
error: %@",toErrorDescription(error)]];
    }
}];
}

```

How do I get the user wallet?

- a. Bind the user - (please refer above bind user section)
- b. Get the wallet

```

// Get the bind user cards.
-(IBAction)getSavedCards:(id)sender{
    // Configure your request here.
    [profileLayer requestPaymentInformationWithCompletionHandler:^(CTSPaymentRes
*paymentInfo, NSError *error) {
        if (error == nil) {
            // Your code to handle success.
            NSMutableString *toastString = [[NSMutableString alloc] init];
            if([paymentInfo.paymentOptions count]){
                [toastString appendString:[self convertToString:[paymentInfo.paymentOptions
objectAtIndex:0]]];
            }
            else{
                toastString =(NSMutableString *) @" no saved cards, please save card first";
            }
            [UIUtility toastMessageOnScreen:toastString];
        } else {

```

This document is meant for PG integration on iOS Platform.

Copyright (c) 2014 Citrus Pay. All rights reserved.

```

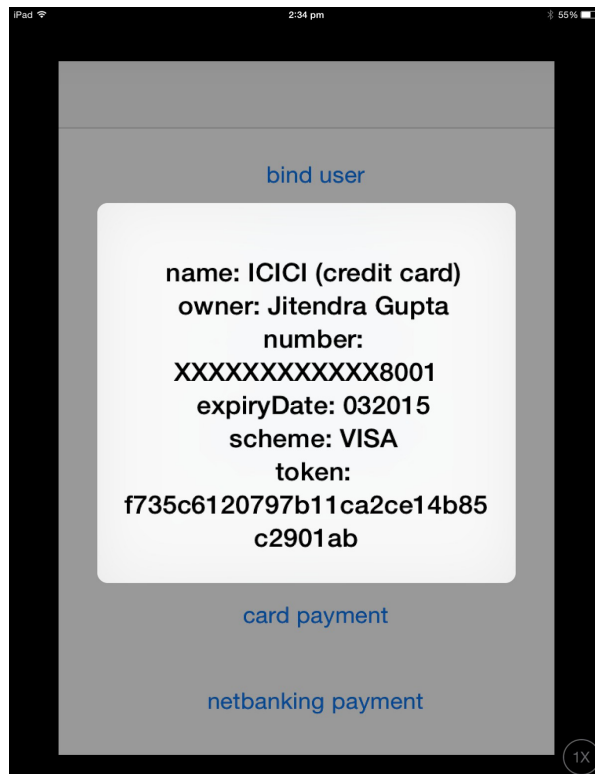
        // Your code to handle error.
        [UIUtility toastMessageOnScreen:[NSString stringWithFormat:@" couldn't find saved
cards \nerror: %@",[error description]]];
    }
    }];
}

```

Where's the token?

When you call **requestPaymentInformationWithCompletionHandler**, in completion handler you are given **CTSPaymentProfile** which contains tokens for all the cards that user has saved

refer **-(IBAction)getSavedCards:(id)sender** of **SimpleViewController.h**



It contains the last 4 digit of the card and the token value. You need to pass this token value along with the cvv to make a tokenized payment.

This document is meant for PG integration on iOS Platform.

Copyright (c) 2014 Citrus Pay. All rights reserved.

What's the return url and where do I set it?

We let you know what happened with your transaction on the return url. You set it inside your bill generator. It's readymade and available in PHP, JAVA, .NET. You can copy it and host it on your server and set it inside your bill generator.

How do I get back to the app after payment?

After you have opened Return Url in [UIWebView](#) and payments are done we redirect you to return url you posted to us, now you need to get back to your app from [UIWebView](#).

To do that you need to implement below delegate,

- (void)webViewDidFinishLoad:(UIWebView*)webView

And add code like following, you can refer **WebViewController.h** from sample app.

```
- (void)webViewDidFinishLoad:(UIWebView*)webView {  
    // Stop the indicator's once the page loaded.  
    [indicator stopAnimating];  
  
    NSDictionary *responseDict = [CTSUtility getResponseIfTransactionIsComplete:webView];  
    if(responseDict){  
        // <responseDict> contains all the information related to transaction.  
        [self transactionComplete:responseDict];  
    }  
}
```

List of banks Available and their CIDs.

Note that every bank may not be available to you since different banks have different merchant acceptance policy.

Sr. No.	Bank Name	Bank CID
1	AXIS Bank	CID002
2	Bank of India	CID019
3	Central Bank of India	CID023
4	CITI Bank	CID003
5	Corporation Bank	CID025
6	DEUTSCHE Bank	CID006
7	Federal Bank	CID009
8	HDFC Bank	CID010
9	ICICI Bank	CID001
10	IDBI Bank	CID011
11	Indian Bank	CID008
12	Indian Overseas Bank	CID027
13	SBI Bank	CID005
14	State Bank of Bikaner and Jaipur	CID013
15	State Bank of Hyderabad	CID012
16	State Bank of Mysore	CID014
17	State Bank of Travancore	CID015
18	Union Bank Of India	CID007
19	United Bank of India	CID041
20	Vijaya Bank	CID042
21	YES Bank	CID004
22	Karnataka Bank	CID031
23	State Bank of Patiala	CID043
24	Bank of Maharashtra	CID021
25	Kotak Mahindra Bank	CID033
26	City Union Bank	CID024
27	PNB Retail	CID044
28	PNB Corporate	CID036
29	ING VYSA	CID0029
30	Bank Of Baroda	CID0046

This document is meant for PG integration on iOS Platform.

Copyright (c) 2014 Citrus Pay. All rights reserved.