



Buy Everywhere, Pay Anywhere

Mobile SDK Specifications for iOS (Version 4.0.4)

Last updated January 23, 2015

MOLPay Sdn Bhd (948015-X)
B-13-3A, Jalan Multimedia 7/AH,
CityPark, i-City, 40000 Shah Alam,
Selangor Darul Ehsan, Malaysia.



+603-55218438



+603-55218437



sales@molpay.com



www.molpay.com

Table of Contents

1. PREREQUISITE	2
2. SYSTEM REQUIREMENTS	2
3. INTEGRATE MOLPAY SDK	2
3.1.CONFIGURE XCODE PROJECT	3
3.2.CONFIGURE MOLPAY SDK	4
3.3.REQUEST PAYMENT RESULT	8
4. MOLWALLET INTEGRATION	10
5. SECURE MODE	12
5.1. INTRODUCTION	12
5.2. IMPLEMENTATION	12
5.3. REQUEST PAYMENT RESULT IN SECURE MODE.....	12
6. LIST OF CHANNELS CODE AND SUPPORTED CURRENCY CODE	13
7. MOLPAY SDK SCREENSHOTS	14

1. Prerequisite

Please provide the information below to verify merchant identity from MOLPay Server in order to use the payment channel:

- merchant ID
- verifykey
- appname
- username
- password

All the values above are in one word without whitespace.

The information should be request from MOLPay support team and the email address is support@molpay.com

2. System Requirements

- Ready for iOS 7 and above
- Minimum iOS version 6 and above. Older version support can be provided on request for custom make
- iOS based devices that support iOS version 4.3

3. Integrate MOLPay SDK

This section explains details on how to integrate MOLPay SDK. MOLPay SDK is a framework that will be added to the Xcode project.

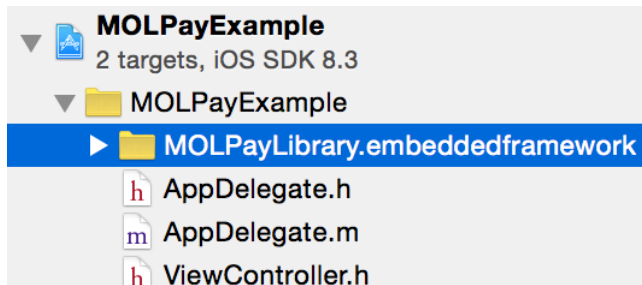
The flow of integration:

- 1) Configure Xcode project (Adding the library & bundles).
- 2) Configure MOLPay SDK.
- 3) Merchant app pass buyer's information to the SDK
- 4) SDK will handle the payment process.
- 5) After payment is completed, SDK will return the result for transaction via delegate.

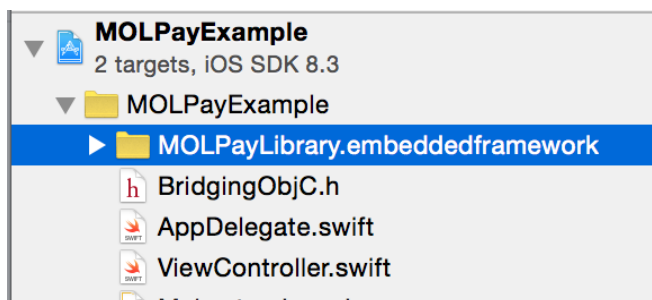
3.1. Configure Xcode Project

- Drag and Drop *MOLPayLibrary.framework* and *MOLPay.bundle* to your workspace.

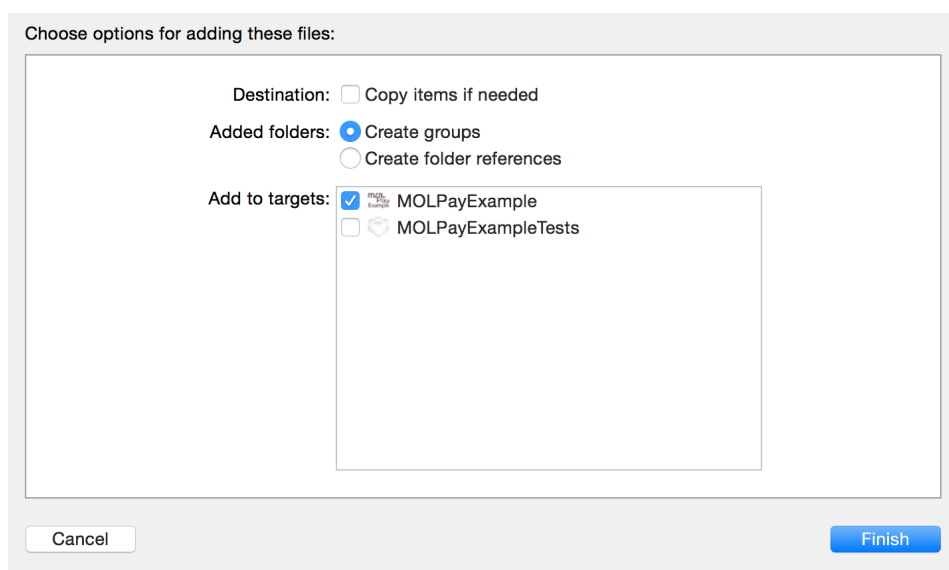
OBJECTIVE-C



SWIFT



- Uncheck the box “*Copy items if needed*” then press *finish* button on bottom right. This will reduce space in your App.



3.2.Configure MOLPay SDK

- **Import Library**

Import MOLPay SDK Library.

OBJECTIVE-C

```
#import <MOLPayLibrary/MOLPayLib.h>
```

SWIFT

Since MOLPay SDK is build on objective-C platform, we need to create a bridging header to import MOLPay SDK.

Create a bridging header yourself by choosing
File > New > File > (iOS or OS X) > Source > Header File.

After created, add the following code
(In MOLPay Swift Example, look BridgingObjC.h file)

```
#import <MOLPayLibrary/MOLPayLib.h>
```

- **Setup delegate.**

Add delegate at header file

OBJECTIVE-C

```
@interface ViewController () <MOLPayDelegate>
```

SWIFT

```
class ViewController: UIViewController,MOLPayDelegate{
```

- **Prepare payment details**

OBJECTIVE-C

```
NSDictionary * paymentRequestDict =  
@{  
    mp_username: (Type: NSString),  
    mp_password: (Type: NSString),  
    mp_merchant_ID: (Type: NSString),  
  
    mp_app_name : (Type: NSString),  
    mp_verification_key: (Type: NSString),  
  
    mp_order_ID: (Type: NSString),  
    mp_amount : (Type: Float),  
  
    mp_currency: (Type: NSString),  
    mp_channel : (Type: NSString),  
    mp_country : (Type: NSString),  
};
```

```
mp_bill_name : (Type: NSString),
mp_bill_email : (Type: NSString),
mp_bill_mobile: (Type: NSString),
mp_bill_description: (Type: NSString),

mp_is_escrow: (Type: BOOL),
mp_channel_editing: (Type: BOOL),
mp_editing_enabled: (Type: BOOL),
mp_debug_mode: (Type: BOOL),
mp_navigation_bar_tint_color: (Type: UIColor),

mp_is_secure: (Type: BOOL), //Please refer chapter 6

mp_url_scheme: (Type: NSString)
};
```

SWIFT

```
var paymentDetails =
[
    mp_username: (Type: NSString),
    mp_password: (Type: NSString),
    mp_merchant_ID: (Type: NSString),

    mp_app_name : (Type: NSString),
    mp_verification_key: (Type: NSString),

    mp_order_ID: (Type: NSString),
    mp_amount : (Type: Float),

    mp_currency: (Type: NSString),
    mp_channel : (Type: NSString),
    mp_country : (Type: NSString),

    mp_bill_name : (Type: NSString),
    mp_bill_email : (Type: NSString),
    mp_bill_mobile: (Type: NSString),
    mp_bill_description: (Type: NSString),

    mp_is_escrow: (Type: BOOL),
    mp_channel_editing: (Type: BOOL),
    mp_editing_enabled: (Type: BOOL),
    mp_debug_mode: (Type: BOOL),
    mp_navigation_bar_tint_color: (Type: UIColor),

    mp_is_secure: (Type: BOOL), //Please refer chapter 6

    mp_url_scheme: (Type: NSString)
]
```

• Allocate class

OBJECTIVE-C

```
MOLPayLib *molpay= [[MOLPayLib alloc] initWithTarget:self
                    andPaymentDetails:paymentRequestDict];
```

SWIFT

```
var molpay: MOLPayLib = MOLPayLib(target: self,
                                   andPaymentDetails: paymentDetails)
```

- **Load the SDK**

OBJECTIVE-C

```
[molpay loadMOLPaySDK];
```

SWIFT

```
molpay.loadMOLPaySDK()
```

- **Delegate return result**

OBJECTIVE-C

```
- (void)paymentRequestSuccessWithPaymentDetails:(NSDictionary
*)paymentDetailsDict{
    // Display Result
}

- (void)paymentRequestFailedWithErrorDescription:(NSDictionary
*)errorDescription{
    // Display Result
}
```

SWIFT

```
func paymentRequestSuccessWithPaymentDetails(
    paymentDetailDict: [NSObject : AnyObject]!) {
    // Display Result
}

func paymentRequestFailedWithErrorDescription(
    errorDescription: [NSObject : AnyObject]!) {
    // Display Result
}
```

- **Note**

- Please refer chapter 5 for full list of channel.
- Debug mode should be disable before release the app. It might slow the app.
- Enable editing will let the user edit name, email, mobile number and description.
- URL Scheme is needed for MOLWallet channel. (For more info please refer to chapter 3.4.
- Example of success return result

```
{
  "txn_ID": "0123456",
  "paydate": 1411370328,
  "amount": "100.00",
  "status_code": "22",
  "msgType": "C6"
}
```

- Example of failed return result

```
{  
  "error_desc": "User exit the payment page",  
  "status": "exit"  
}
```

- Transaction status:

Status	Description
00	Success
11	Failed
22	Pending

- Field included in paymentDetailsDict:

Key	Value type
txn_ID	Numeric (Integer)
paydate	Numeric (Integer)
amount	Numeric (Float)
status_code	Numeric (Integer)
channel	String
err_desc	String
app_code	String
msgType	String

3.3.Request payment result

- Request by transaction ID

OBJECTIVE-C

```
NSDictionary * request = @{
    mp_transaction_id: (Type: NSString),
    mp_merchant_ID(Type: NSString),
    mp_amount(Type: NSString),
    mp_verification_key(Type: NSString),
    mp_receipt_enabled:(Type: BOOL)
};

MOLPayLib * molpay = [[MOLPayLib alloc] initWithTarget:self
andPaymentDetails:nil];

[molpay requestPaymentResult: request];
```

SWIFT

```
var request = [
    mp_transaction_id: (Type: NSString),
    mp_merchant_ID(Type: NSString),
    mp_amount(Type: NSString),
    mp_verification_key(Type: NSString),
    mp_receipt_enabled:(Type: BOOL)
]

var molpay: MOLPayLib = MOLPayLib(target: self,
andPaymentDetails: paymentDetails)

molpay.requestPaymentResult(request)
```

- Request by order ID:

OBJECTIVE-C

```
NSDictionary * request = @{
    mp_order_ID: (Type: NSString),
    mp_merchant_ID(Type: NSString),
    mp_amount(Type: NSString),
    mp_verification_key(Type: NSString),
    mp_receipt_enabled:(Type: BOOL)
};

MOLPayLib * molpay = [[MOLPayLib alloc] initWithTarget:self
andPaymentDetails:nil];

[molpay requestPaymentResult: request];
```

SWIFT

```
var request = [
    mp_order_ID: (Type: NSString),
    mp_merchant_ID(Type: NSString),
    mp_amount(Type: NSString),
    mp_verification_key(Type: NSString),
    mp_receipt_enabled:(Type: BOOL)
]

var molpay: MOLPayLib = MOLPayLib(target: self,
                                   andPaymentDetails: paymentDetails)

molpay.requestPaymentResult(request)
```

- **Retrieve result via delegate**

OBJECTIVE-C

```
- (void)resultRequestSuccessWithPaymentDetails:
    (NSDictionary *)paymentDetailsDict{
    // Display Result
}

- (void)resultRequestFailedWithErrorDescription:
    (NSDictionary *)paymentDetailsDict{
    // Display Result
}
```

SWIFT

```
func resultRequestSuccessWithPaymentDetails(
    paymentDetailsDict: [NSObject : AnyObject]!) {
    // Display Result
}

func resultRequestFailedWithErrorDescription(
    paymentDetailsDict: [NSObject : AnyObject]!) {
    // Display Result
}
```

4. MOLWallet Integration

Merchant apps need to specify their own URL scheme and pass it in the paymentRequestDict (Please refer to chapter 3.3.1). URL Scheme will be use to handle callbacks in MOLWallet channel.

• Setup info.plist

Press the add button and type URL types, Xcode give suggestion. Press return/enter.

Xcode will auto generate the content.

Supporting Files	Launch screen interface file base...	String	LaunchScreen
Info.plist	Main storyboard file base name	String	Main
main.m	Required device capabilities	Array	(1 item)
MOLPayExampleTests	Supported interface orientations	Array	(3 items)
Products	Supported interface orientations (...)	Array	(4 items)
	URL types	String	

• Set the URL identifier

In the value for URL identifier of your own. In this example, com.molpay.example.

Supported interface orientations (...)	Array	(4 items)
URL types	Array	(1 item)
Item 0	Dictionary	(1 item)
URL identifier	String	com.molpay.example

• Adding URL scheme

Press the add button, and select URL Schemes.

Item 0 (None)	Dictionary	(2 items)
URL identifier	String	com.molpay.example
Document Role	String	None
Document Icon File Name		
Document Role		
URL identifier		
URL Schemes		

Enter the URL Scheme of your own. In this example, com.molpay.example.

Item U	Dictionary	(2 items)
URL identifier	String	com.molpay.example
URL Schemes	Array	(1 item)
Item 0	String	molpayExample

- **Handling URL Scheme (App communication)**

AppDelegate file (.m file)

```
- (BOOL)application:(UIApplication *)application
    openURL:(NSURL *)url
    sourceApplication:(NSString *)sourceApplication
    annotation:(id)annotation{

    return [MOLPayLib handleOpenURL:url
        sourceApplication:sourceApplication];
}
```

5. Secure Mode

5.1. Introduction

Secure mode was introduced to improve security level of SDK. This mode is aiming to protect the merchant sensitive data such as verify key from hardcoding in merchant's application.

5.2. Implementation

In order to implement this new feature, please enable the following flag when you pass `paymentRequestDict` (see *chapter 3.2 - Prepare payment details*) to the SDK.

```
mp_is_secure: (Type: BOOL),
```

Merchant are encourage to generate the `vcode` & in their server and pass to the SDK through merchant app.

Below is the formula to generate the `vcode`:

MD5 (AMOUNT + MERCHANT ID + ORDER ID + VERIFY KEY + MSG TYPE)

Example in PHP:

```
$vcode = md5( $POST['amount'] . $POST['merchant_id'] .  
$POST['order_id'] . $verifyKey . $POST['msg_type'] );
```

Passing `vcode` to SDK (see *chapter 3.2 - Prepare payment details*):

```
mp_verification_key: vcode(Type: NSString)
```

5.3. Request Payment Result in Secure Mode

In order to request payment result manually, please pass the `vcode` generated previously in chapter 5.2. Please refer to chapter 3.3 for more details.

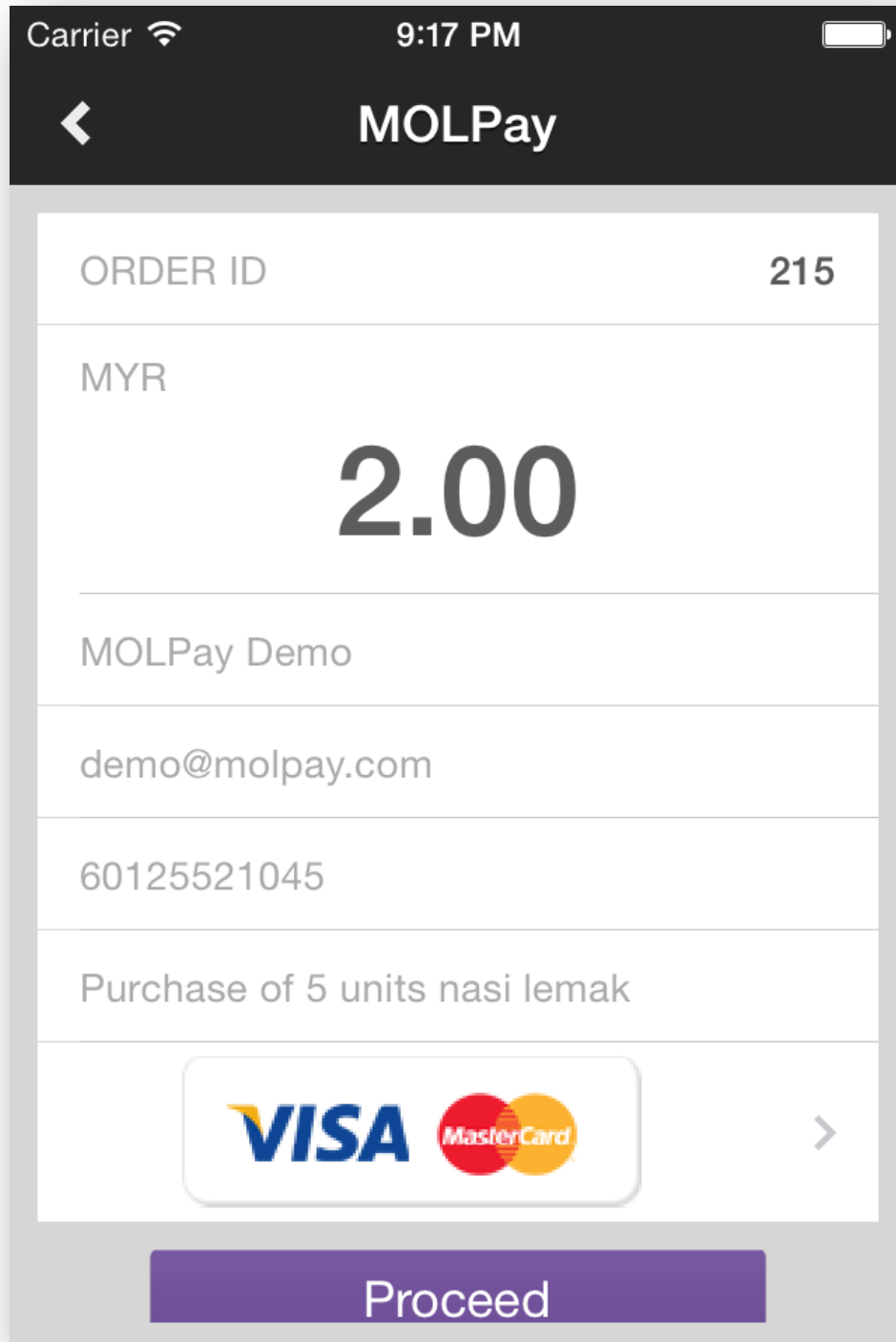
```
mp_verification_key: vcode, //(Type: NSString)
```

6. List of Channels Code and Supported Currency Code

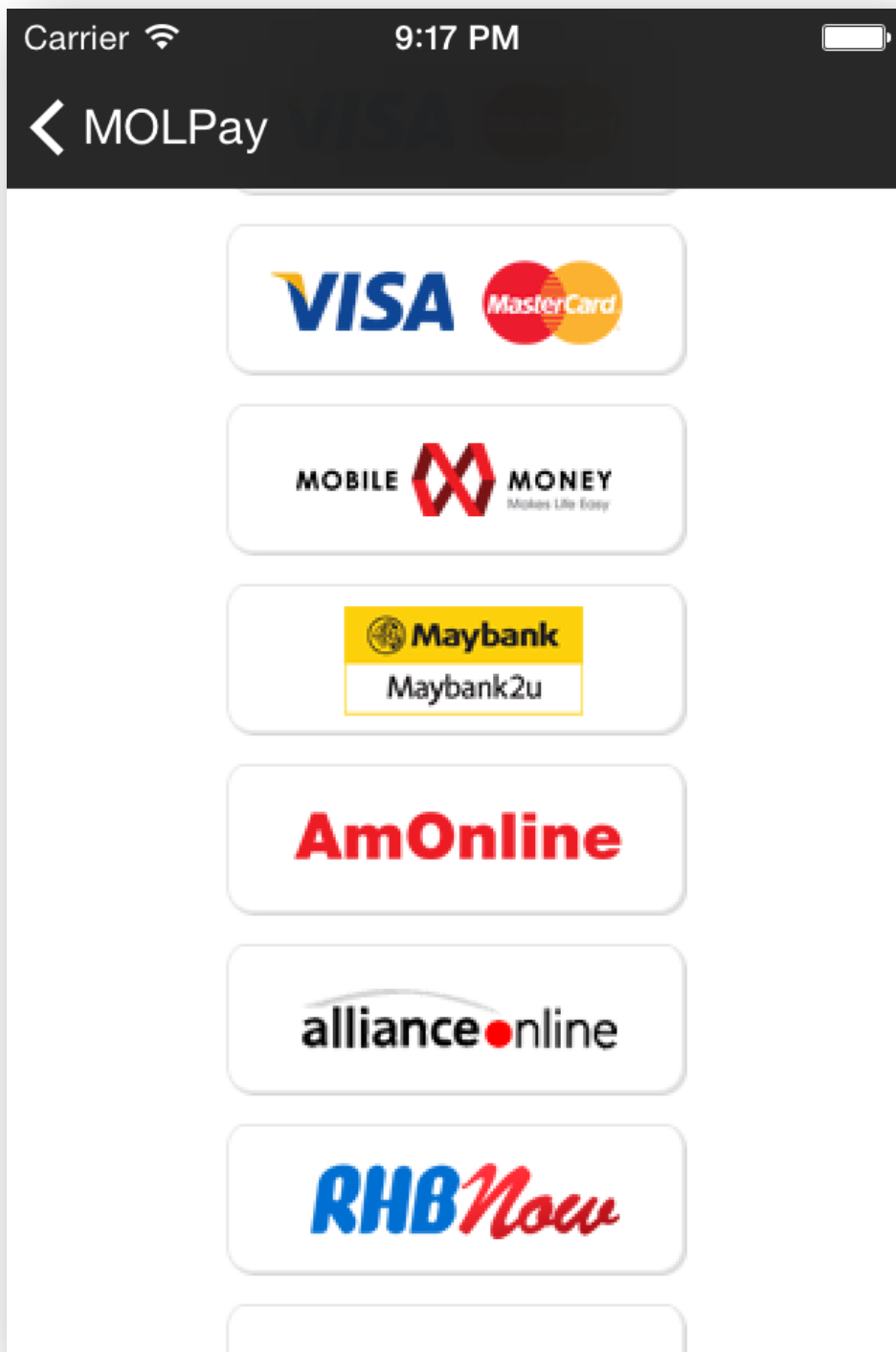
No	Channel Code	Channel Name	Currency Code
1.	credit	Visa / Mastercard	MYR
2.	maybank2u	Maybank2u	MYR
3.	amb	AmOnline	MYR
4.	rhb	RHB Now	MYR
5.	webcash	WEBCASH	MYR
6.	paypal	PayPal	USD, AUD, GBP, CAD, CZK, DKK, EUR, HKD, HUF, ILS, JPY, MYR, MXN, NZD, NOK, PHP, PLN, SGD, SEK, CHF, TWD and THB
7.	alipay	Alipay	MYR, RM, USD, TWD and RMB
8.	hlb	Hong Leong Online	MYR
9.	cimb	CIMB Clicks	MYR
10.	fpx	MEPS FPX	MYR
11.	dragonpay	DragonPay	USD and PHP
12.	enetsD	eNets Debit	SGD
13.	polipayment	Poli Payment	AUD
14.	cash	7-Eleven Cash Retail	MYR
15.	singpost	SingPost	SGD
16.	MOLWallet	MOLWallet	MYR
17.	multi	List all channel	

7. MOLPay SDK Screenshots



- Order details



- **List of Channels**



- Credit / Debit card


Carrier  9:18 PM 

< Back

ORDER ID247

MYR

2.00

41111111111111111111

...

02 / 2016

MALAYSIA >

CIMB BANK BERHAD

Proceed

- **Payment Page**

Carrier
9:19 PM

Exit


CIMB BANK


Secure ePay Code has been sent to your registered mobile phone number. **5.** Please enter the Secure ePay Code to authenticate this payment.

Merchant Name	: Netbder Netbuilder
Amount	: MYR 2.00
Transaction Date	: Fri May 9 2014 21:19:03 GMT+0800
CIMB Bank MasterCard No.	: xxxx xxxx xxx
SecureCode™ Secure ePay Code	: <input type="text"/>

Submit
Cancel

If you do not receive Secure ePay Code within the next few minutes, please click on "Resend Secure ePay Code" button for a new secure ePay Code.

Resend Secure ePay Code

This information is **not shared** with the Merchant.


Please contact our Customer Service Hotline at the back of your card for assistance.