

Version: 4.19.20



HƯỚNG DẪN TÍCH HỢP ZPDK Ver 4.19.20

HƯỚNG DẪN PHẦN ANDROID

Phần tích hợp ZPDK phía android gồm 4 bước như sau:

1. Trong project Android của Merchant, cần import Lib ZPDK của ZaloPay vào (file *.aar) được đính kèm trong lúc gửi.
File -> New -> New Module -> Insert Module JAR/AAR file.
 - 1.1. File -> New -> New Module -> Insert Module JAR/AAR file.
 - 1.2. Chọn file zpdk.aar, đặt tên cho Module.
 - 1.3. Kiểm tra trong file build.gradle của module zpdk có dạng

```
configurations.maybeCreate("default")
artifacts.add("default", file("zpdk-release.aar"))
```

2. Thêm lời gọi hàm khởi tạo ZaloPaySDK trong onCreate của Application:
 - 2.1. Trong đó appld sẽ là appId của ZaloPay đã cấp cho Merchant.

```
@Override
public void onCreate() {
    super.onCreate();
    // khởi tạo ZPDK
    ZaloPaySDK.getInstance().initWithAppId(3);
}
```

3. Gọi hàm thanh toán từ ZaloPaySDK và nhận Callback trở về như sau:

```
// Trước khi gọi thanh toán, cần forward lời gọi onActivityResult tới ZPDK
@Override
public void onActivityResult(Activity activity, int requestCode,
                             int resultCode, Intent data) {
    ZaloPaySDK.getInstance().onActivityResult(requestCode, resultCode, data);
}

// Gọi hàm thanh toán
ZaloPaySDK.getInstance().payOrder(
    getActivity(), zpTranstoken, new MyZaloPayListener()
);

// Implement interface ZaloPayListener để nhận kết quả thanh toán
private static class MyZaloPayListener implements ZaloPayListener {
    @Override
    public void onPaymentSucceeded(String transactionId, String zpTranstoken) {
        Log.d(TAG, "onSuccess: On successful with transactionId: " + transactionId + "zpTranstoken: " + zpTranstoken);
    }
    @Override
    public void onPaymentError(ZaloPayErrorCode errorCode, int paymentErrorCode, String zpTranstoken) {
        Log.d(TAG, String.format("onPaymentError: payment error with [error: %s,
                                paymentError: %d, zpTranstoken: %s]", errorCode, paymentErrorCode, zpTranstoken));
    }
}
```

4. Ngoài ra, cần đăng ký thêm broadcast để nhận callback của ZaloPay.

4.1. Thêm định nghĩa receiver trong AndroidManifest:

```
<receiver
    android:name="vn.zalopay.sdk.MerchantReceiver"
    android:exported="true">
    <intent-filter>
        <action android:name="vn.zalopay.sdk.ZP_ACTION" />
        <category android:name="zlp" />
    </intent-filter>
</receiver>
```

Sample app:

<https://github.com/nghianguyen979/zlp-mer-demo>

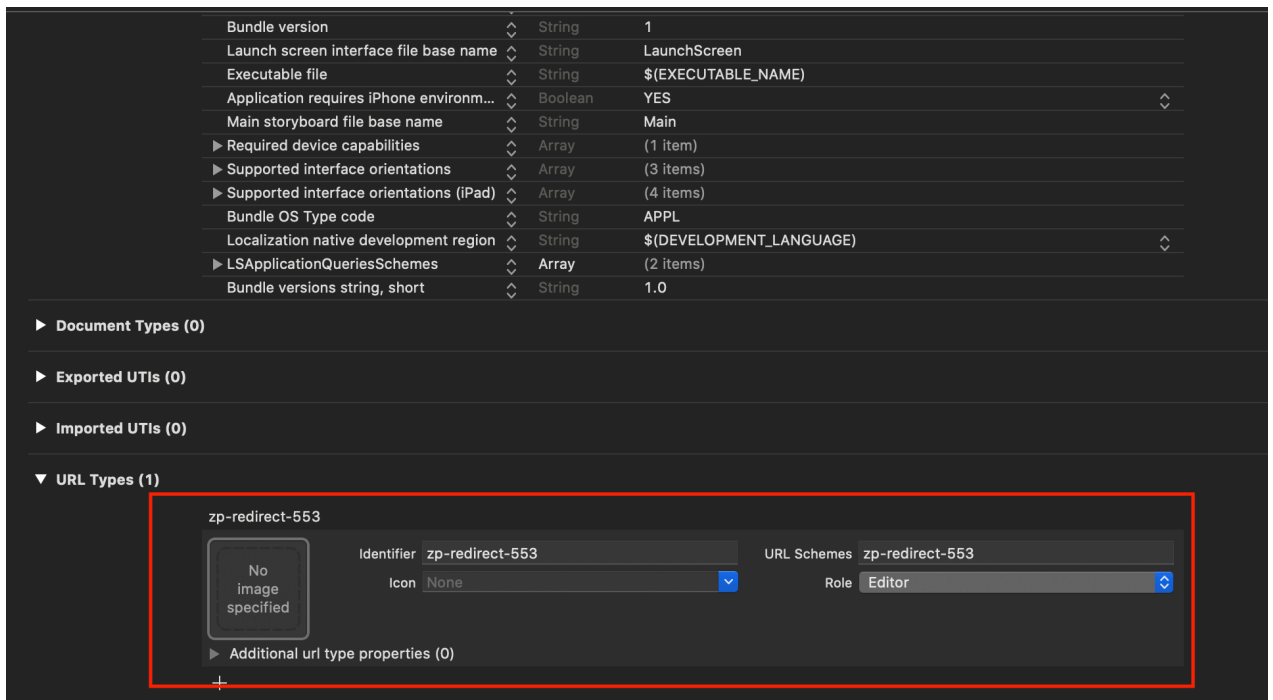
Cách sử dụng sample app để test:

1. Truy cập vào <https://sandbox.zalopay.com.vn/qrcode/>
2. Chọn [QRCode > OpenApp] -> Dùng app sample để scan (sample app lấy transtoken để test).

HƯỚNG DẪN TÍCH HỢP CHO IOS:

1. Tích hợp theo các bước sau đây:

- 1.1. Thêm framework ZPDK.framework vào project
- 1.2. Cần tạo thêm custom URL scheme cho app, với url có định dạng:
zp-redirect-*<appid>*:// (ví dụ: zp-redirect-553://).
 Xem hình minh hoạ như bên dưới.



- 1.3. Thêm cấu hình cho phép khởi tạo ZaloPay từ app, bằng cách thêm giá trị **zalopay.api.v2** và **zalopay** vào key **LSApplicationQueriesSchemes**.

```

    <key>CFBundleURLIconFile</key>
    <string></string>
    <key>CFBundleURLName</key>
    <string>zp-redirect-553</string>
    <key>CFBundleURLSchemes</key>
    <array>
      <string>zp-redirect-553</string>
    </array>
  </dict>
</array>
<key>CFBundleVersion</key>
<string>1</string>
<key>LSApplicationCategoryType</key>
<string></string>
<key>LSApplicationQueriesSchemes</key>
<array>
  <string>zalopay</string>
  <string>zalopay.api.v2</string>
</array>
<key>LSRequiresiPhoneOS</key>
<true/>
<key>UILaunchStoryboardName</key>
<string>LaunchScreen</string>
<key>UIMainStoryboardFile</key>
<string>Main</string>
<key>UIRequiredDeviceCapabilities</key>
<array>
  <string>armv7</string>

```

2. Khởi tạo ZDPK cho App:

2.1. Trong AppDelegate.swift, thêm lời gọi tới ZPKD để xử lý việc trao đổi dữ liệu giữa ZaloPay và app

```

func application(_ application: UIApplication, didFinishLaunchingWithOptions launchOptions: [UIApplication.LaunchOptionsKey: Any]?) -> Bool {
    ZaloPaySDK.sharedInstance()?.initWithAppId(553)//Khởi tạo ZPKD
    return true
}

//gọi tới ZPKD để xử lý việc trao đổi dữ liệu giữa ZaloPay và app
func application(_ app: UIApplication, open url: URL, options: [UIApplication.OpenURLOptionsKey : Any] = [:]) -> Bool {
    ZaloPaySDK.sharedInstance().application(app, open: url, sourceApplication: "vn.com.vng.zalo", annotation: nil)
}

```

Lưu ý: Chỉ gọi hàm này vì ZPKD hiện đang check sourceApplication có đúng là của ZaloPay hay không.

3. Gọi hàm thanh toán:

3.1. Gọi hàm để bắt đầu thanh toán

```

ZaloPaySDK.sharedInstance()?.delegate=self
ZaloPaySDK.sharedInstance()?.payOrder(self.zpTransToken.text)

```

3.2. Xử lý kết quả trả về:

Version: 4.19.20

```
func zalopayCompleteWith(_ errorCode: ZPErrorCode, transactionId: String!, zpTranstoken zptranstoken: String!) {
    print("----- errorCode -----: \(errorCode.rawValue)")

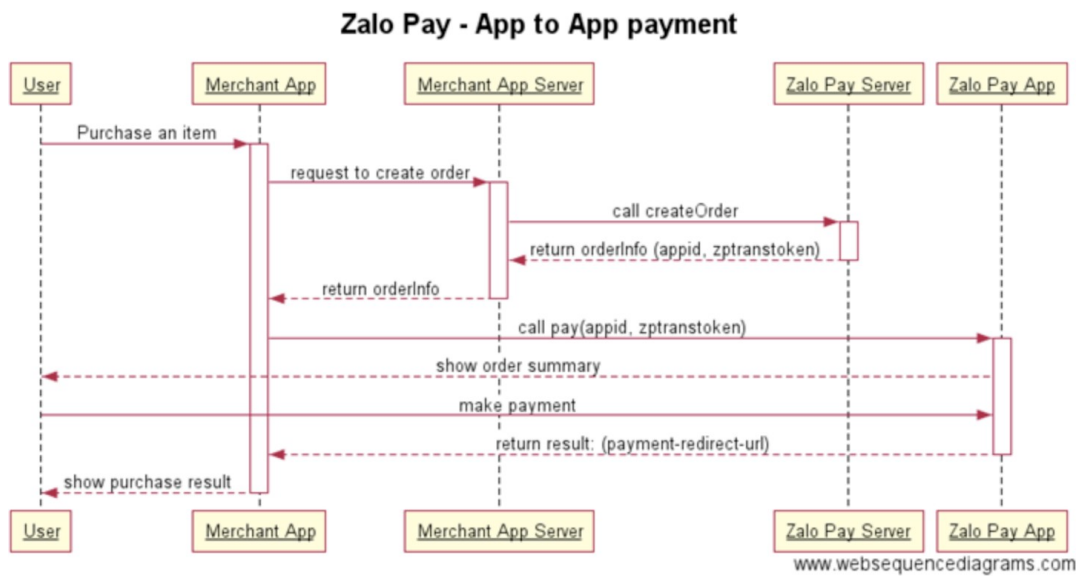
    if errorCode.rawValue == 1 {
        lbResult.text = "Thanh toán thành công"
    }else{
        lbResult.text = "Thanh toán thất bại, error: \(errorCode.rawValue)"
        if errorCode.rawValue == -1{//not install
            //installProduction()
            installSandbox()
        }
    }
}

}
```

ZPDK User Guide © Copyright by ZaloPay

SƠ ĐỒ XỬ LÝ TUẦN TỰ

- Sơ đồ xử lý tuần tự trường hợp thanh toán có cài app ZaloPay:



- Sơ đồ tuần tự trường hợp thanh toán không có cài đặt app ZaloPay:

