**SỞ THÔNG TIN VÀ TRUYỀN THÔNG THÀNH PHỐ HỒ CHÍ MINH**

**--------------**

**MÔ HÌNH LIÊN THÔNG NỀN TẢNG HCM LGSP TPHCM**

**DỊCH VỤ TÍCH HỢP CỔNG THANH TOÁN**

**Phiên bản : 1.1**

**TP.HCM, 8/2019**

**BẢNG GHI NHẬN THAY ĐỔI TÀI LIỆU**

| **Ngày thay đổi** | **Vị trí thay đổi** | **Phiên bản** | **Mô tả thay đổi** |
| --- | --- | --- | --- |
| 15/12/2020 | V.1.0 | 1.1 | API thanh toán : thêm AccountCode , chuỗi checksum . |

**MỤC LỤC**

[THUẬT NGỮ VÀ TỪ VIẾT TẮT 2](#_Toc13487190)

[DANH MỤC HÌNH VẼ, BẢNG BIỂU 3](#_Toc13487191)

[I. TỔNG QUAN 4](#_Toc13487192)

[1.1. Mô tả nghiệp vụ 4](#_Toc13487193)

[1.2. Mục đích tài liệu 4](#_Toc13487194)

[1.3. Phạm vi và đối tượng áp dụng 4](#_Toc13487195)

[1.4. Thông tin kết nối 4](#_Toc13487196)

[1.5. Thông tin dịch vụ E-payment 4](#_Toc13487197)

[II. MÔ HÌNH KẾT NỐI 5](#_Toc13487198)

[2.1. Mô hình kết nối đồng bộ thông tin thanh toán 5](#_Toc13487199)

[2.2. Quá trình trao đổi gói tin 5](#_Toc13487200)

[III. ĐẶC TẢ API ĐỒNG BỘ THÔNG TIN THANH TOÁN 6](#_Toc13487201)

[3.1. Thông tin kết nối 6](#_Toc13487202)

[3.2. API đồng bộ thông tin thanh toán trực tuyến 7](#_Toc13487203)

[IV. CODE EXAMPLE 8](#_Toc13487204)

[4.1. Code mẫu .Net 8](#_Toc13487205)

[4.2. Code mẫu java 10](#_Toc13487206)

[V. Tham khảo 11](#_Toc13487207)

# THUẬT NGỮ VÀ TỪ VIẾT TẮT

| STT | Thuật ngữ | Ý Nghĩa |
| --- | --- | --- |
|  | LGSP | Nền tảng tích hợp và chia sẻ dữ liệu |
|  | CSDL | Cơ sở dữ liệu |
|  | HCM | Thành phố Hồ Chí Minh |
|  | JSON | JavaScript Object Notation: một kiểu dữ liệu mở trong JavaScript |

# DANH MỤC HÌNH VẼ, BẢNG BIỂU

| STT | Hình vẽ/ Bảng | Ý nghĩa |
| --- | --- | --- |
|  | Hình 1 | Mô hình kết nối đồng bộ thông tin thanh toán |
|  | Hình 2 | Quy trình trao đổi gói tin |
|  | Bảng 1 | Thông tin kết nối |
|  | Bảng 2 | Tham số truyền vào chuỗi json của body |
|  | Bảng 3 | Tham số trả về của chuỗi json |

# TỔNG QUAN

## Mô tả nghiệp vụ

* Phần mềm của đơn vị khi có nhu cầu yêu cầu thanh toán trực tuyến lên HCM LGSP
* HCM LGSP nhận yêu cầu thanh toán của đơn vị kết nối và sẽ gửi yêu cầu thanh toán đến các ví điện tử / cổng thanh toán .
* Sau khi HCM LGSP nhận được kết quả thanh toán sẽ gửi trả thông tin thanh toán về cho các đơn vị kết nối .

## Mục đích tài liệu

* 1. Tài liệu này nhằm hướng dẫn đơn vị kết nối thanh toán trực tuyến
  2. Phạm vi và đối tượng áp dụng
* Tài liệu này được sử dụng bởi các kỹ thuật viên của các đơn vị cần kết nối thanh toán trực tuyến

## Thông tin kết nối

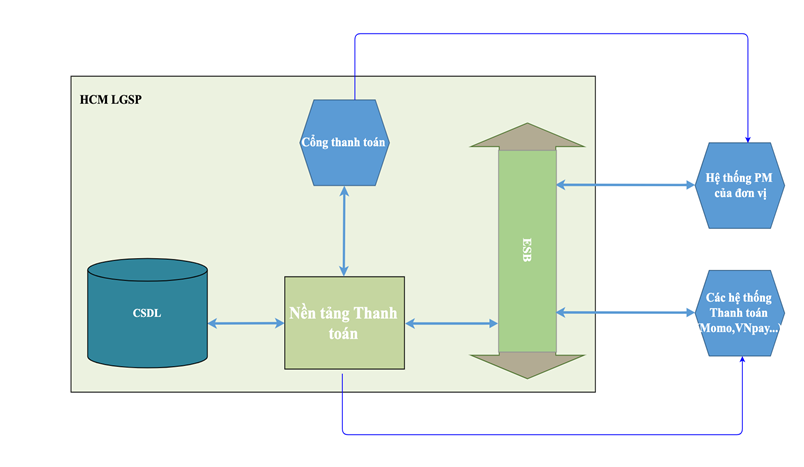
* Đơn vị cần kết nối liên hệ sở Thông tin và Truyền thông để được cung cấp thông tin kết nối bao gồm:
  + PartnerCode
  + AccessKey
  + SecretKey

## Thông tin dịch vụ E-payment

* Đơn vị kết nối với cổng thanh toán thành phố cần ký hợp đồng và cung cấp thông tin tài khoản cho sở Thông tin và Truyền thông các nhà cung cấp dịch vụ sau:
  + MoMo
  + VnPay
  + NaPas
  + Viettel pay
  + Ngân Lượng

# MÔ HÌNH KẾT NỐI

## Mô hình kết nối đồng bộ thông tin thanh toán



*Hình 1: Mô hình kết nối thanh toán trực tuyến*

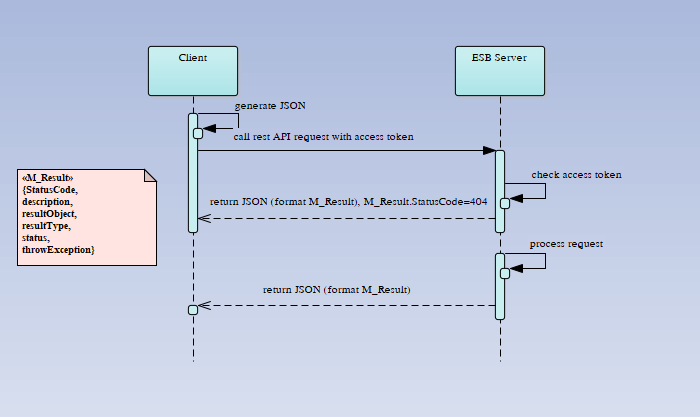
Miêu tả sơ lược quy trình kết nối thanh toán:

Bước 1: Người dùng thao tác trên Website dịch vụ, chọn dịch vụ cần thanh toán và chọn thanh toán trực tuyến.

Bước 2: Thông qua ESB, website dịch vụ gửi thông tin thanh toán của người dùng sang Cổng thanh toán. Người dùng lựa chọn một hình thức thanh toán trong danh sách các cổng thanh toán và ví điện tử cung cấp bởi cổng thanh toán .

Bước 3: Cổng thanh toán gửi yêu cầu thanh toán đến cổng thanh toán/ví điện tử mà người dùng chọn. Người dùng thao tác các bước thanh toán trên cổng thanh toán/ví điện tử đã chọn.  
Sau khi người dùng hoàn tất thanh toán, Cổng thanh toán trả kết quả thanh toán về Website dịch vụ.

## Quá trình trao đổi gói tin



*Hình 2: Quy trình trao đổi gói tin*

* Bước 1: Phần mềm của đơn vị đóng gói dữ liệu về dạng JSON.
* Bước 2: Phần mềm của đơn vị gọi đến Nền tảng HCM LGSP thông qua api kèm theo access token.
* Bước 3: Nền tảng HCM LGSP kiểm tra access token của đơn vị. Nếu thất bại sẽ trả về gói tin JSON lỗi cho phần mềm của đơn vị.
* Bước 4: Nền tảng HCM LGSP xử lý dữ liệu và trả về gói tin JSON kết quả cho phần mềm đơn vị.

# ĐẶC TẢ API YÊU CẦU THANH TOÁN

## Thông tin kết nối

* Đơn vị đang trong giai đoạn thử nghiệm sẽ dùng hệ thống test theo địa chỉ apdapter Nền tảng HCM thử nghiệm
* Đơn vị khi có thông báo vận hành chính thức sẽ dùng hệ thống chính theo địa chỉ apdapter Nền tảng HCM chính

|  |  |  |
| --- | --- | --- |
| **Kiểu** | **Diễn giãi** | **Ví dụ** |
| Địa chỉ adapter Nền tảng HCM chính | Địa chỉ cổng dịch vụ dự kiến chính thức | https://hcmlgsp.tphcm.gov.vn |
| Địa chỉ adapter Nền tảng HCM thử nghiệm | Địa chỉ cổng dịch vụ thử nghiệm | https://hcmesb-test.tphcm.gov.vn/payment |
| Authorization/Token (Thử nghiệm) | Khai báo trong Header, Là chuỗi | ewogICJBY2Nlc3NLZXkiOiIwNGVlZWM5NWJhMzI0YWEzOTQ1YmVkOTUwZjc5YTNkOCIsCiAgIlNlY3JldEtleSI6Ik1EUmxaV1ZqT1RWaVlUTXlOR0ZoTXprME5XSXRaV1E1TlRCbU56bGhNMlE0IiwKICAiQXBwTmFtZSI6ImhjbV9zc29fYWRtaW4iLAogICJQYXJ0bmVyQ29kZSI6ImhjbV90ZXN0X3NzbyIsCiAgIlBhcnRuZXJDb2RlQ3VzIjoiaGNtX3Rlc3Rfc3NvIgp9 |
| Authorization/Token (Chính thức ) | Khai báo trong Header, Là chuỗi | ewoiQWNjZXNzS2V5IjoiODkyNjA1NzI4N2FjYTdlZTBiNzdjNGYwMWZjMmU0ZTYiLAoiU2VjcmV0S2V5IjoiWWpjM0RreU5qQTFOekk0TjJGallUZGxaVEFZelJtTURGbVl6SmxOR1UyTyIsCiJBcHBOYW1lIjogImZpc19lc2IiLAoiUGFydG5lckNvZGUiOiAiMDAwLjAwLkZJUy5FU0IiLAoiUGFydG5lckNvZGVDdXMiOiAiMDAwLjAwLkZJUy5FU0IiCn0= |

*Bảng 1: bảng thông tin kết nối*

## API thanh toán trực tuyến

* Đường dẫn: /paygate
* Method: POST

#### Đầu vào:

* + Header: Authorization
  + Body: chuỗi json chứa thông tin truyền vào

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tham số | Thuộc tính tham số | Mô tả | Lưu ý |
| 1 | partnerCode | String | partnerCode của tài khoản đơn vị khai thác | Bắt buộc |
| 2 | accessKey | String | Chuỗi mã hóa xác định quyền truy cập | Bắt buộc |
| 3 | returnUrl | String | Địa chỉ trả về khi thực hiện thanh toán xong | Bắt buộc |
| 3 | orderId | String | Mã đơn hàng cần thanh toán | Bắt buộc |
| 4 | amount | Int | Tổng số tiền cần thanh toán | Bắt buộc |
| 5 | orderInfo | String | Thông tin mô tả nội dung thanh toán | Bắt buộc |
| 6 | requestCode | String | Mã yêu cầu thanh toán ( Mã này dùng để tham chiếu đến cổng thanh toán ) | Bắt buộc |
| 7 | ipAddress | String | Địa chỉ ip khách hàng thanh toán | Bắt buộc |
| 8 | serviceCode | String | Mã dịch vụ cần thanh toán | Bắt buộc |
| 10 | accountCode | String | Mã tai khoản cổng thanh toán cấp | Bắt buộc |
| 9 | checksum | String | Mã đảm bảo toàn vẹn dữ liệu. Dùng giao thức Sha256 để mã hóa các thông tin gồm : secretKey +partnerCode + accessKey+ orderId + orderInfo +requestCode + amount+ serviceCode + returnUrl + ipAddress+ accountCode | Bắt buộc |

*Bảng 2: tham số truyền vào chuỗi json của body*

Ví dụ:

|  |
| --- |
| Chuỗi json |
| {  "partnerCode": "000.00.18.H29",  "accessKey": " f6047c5b34702d1edd05926c421317f62a58edc3e1c06ce74820d",  "amount": 40000,  "orderId": "SBN\_100012",  "orderInfo": "thanh toan tien dien thoai",  "serviceCode":"hcm\_dichvucong",  "requestCode": "12357851",  "returnUrl": "http://paydate.com.vn/return",  "ipAddress":"120.72.114.122",  "accountCode":"payment\_test ",  "checksum":"3B2C7C400EB89625A82B9F92B82863F4E997A9DBDEA726AF7FBB4F2DBF6C2ABD "  } |
|  |

#### Đầu ra:

* + Chuỗi json chứa thông tin trả về , resultObject là true hoặc false

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tham số | Thuộc tính tham số | Mô tả |
| 1 | error\_code | String | Mã lỗi thực hiện yêu cầu:  SUCCESSFUL: thành công  FAILED: thất bại,  PARAM\_ERROR: Dữ liệu gửi lên thiếu nội dung  SIGNTURE\_WRONG: Dữ liệu không toàn vẹn  ORDER\_EXITS: Giao dịch đã tồn tại |
| 2 | error\_message | String | Thông tin error message |
| 3 | data | JSON | Địa chỉ trả về đường dẫn thanh toán |

*Bảng 3: tham số trả về của chuỗi json*

Ví dụ:

|  |  |
| --- | --- |
| Thành công | Lỗi |
| {  "error\_code":"SUCCESSFUL",  "error\_message":"",  "data":{  "url": "[http://103.9.0.215/paygate/napas/payment? partnerCode=000.00.18.H29 &access-Key=356019473b1ed272d9ca,75780856 &requestCode=1257815 &orderId=Test &amount=30000 &signature= b761088b86da6335befd7d7583e602f211751410 ba4ce842b8b8fc718a9540fc](http://103.9.0.215/paygate/napas/payment?partnerCode=000.00.18.H29&access-Key=356019473b1ed272d9ca,75780856&requestCode=1257815&orderId=Test&amount=30000&signature=b761088b86da6335befd7d7583e602f211751410ba4ce842b8b8fc718a9540fc)"  } | {  "error\_code": "FAILED",  "error\_message": "",  "data": null  } |

## API nhận kết quả thanh toán

* Khi gửi yêu cầu thanh toán, website dịch vụ phải gửi URL của API nhận kết quả thanh toán này trong trường dữ liệu returlURL. Sau khi hoàn tất thanh toán, cổng thanh toán sẽ redirect ngược về returnURL của website dịch vụ.
* Đây là API do đơn vị quản lý website dịch vụ xây dựng.
* Method: PUT

#### Đầu vào:

* + Body: chuỗi json chứa thông tin truyền vào

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tham số | Thuộc tính tham số | Mô tả | Lưu ý |
| 1 | paygate | String | Tên cổng thanh toán | Bắt buộc |
| 2 | payTransId | String | Mã giao dịch tại đơn vị chấp nhận thanh toán | Bắt buộc |
| 3 | orderId | String | Mã hóa đơn yêu cầu thanh toán | Bắt buộc |
| 3 | amount | Int | Số tiền thanh toán | Bắt buộc |
| 4 | orderInfo | String | Thông tin hóa đơn thanh toán | Bắt buộc |
| 5 | payDate | String | Thời gian khách hàng thanh toán. Định dạng:  yyyyMMddHHmmss | Bắt buộc |
| 6 | errorCode | String | Mã lỗi kết quả thanh toán  -00: Thành công  -#00: thất bại | Bắt buộc |
| 7 | type | String | Loại  Pay: thanh toán  Refund: Hoàn trả | Bắt buộc |
| 8 | checksum | String | Mã đảm bảo toàn vẹn dữ liệu. Dùng giao thức Sha256 để mã hóa các thông tin gồm : “PAYHCM1.0”+ paygate + orderId + amount + payDate +type+ ordeInfo + payTransId + errorCode | Bắt buộc |

*Bảng 2: tham số truyền vào chuỗi json của body*

Ví dụ:

|  |
| --- |
| Chuỗi json |
| {  "amount": 40000,  "orderId": "SBN\_100012",  "orderInfo": "thanh toan tien dien thoai",  "payTransId": "1258485",  " payDate":" 20191212161254”,  "paygate":"momo”,  "errorCode":" 00”,  "type":" pay”,  "check-sum":"3B2C7C400EB89625A82B9F92B82863F4E997A9DBDEA726AF7FBB4F2DBF6C2ABD "  } |

#### Đầu ra:

* + Chuỗi json chứa thông tin trả về , resultObject là true hoặc false

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tham số | Thuộc tính tham số | Mô tả |
| 1 | error\_code | String | Mã lỗi thực hiện yêu cầu:  SUCCESSFUL: thành công  FAILED: thất bại |
| 2 | error\_message | String | Thông tin error message |

*Bảng 3: tham số trả về của chuỗi json*

Ví dụ:

|  |  |
| --- | --- |
| Thành công | Lỗi |
| {  "error\_code": "SUCCESSFUL",  "error\_message": "Thành công",  } | {"error\_code": " FAILED",  "error\_message": "Thất bại",  } |

## API hoàn trả

* Đường dẫn: /refund
* Method: POST

#### Đầu vào:

* + Header: Authorization
  + Body: chuỗi json chứa thông tin truyền vào

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tham số | Thuộc tính tham số | Mô tả | Lưu ý |
| 1 | partnerCode | String | Mã đơn vị | Bắt buộc |
| 2 | accessKey | String | Chuỗi mã hóa xác định quyền truy cập | Bắt buộc |
| 3 | transactionNo | String | Mã giao dịch( Mã thanh toán của hóa đơn ) | Bắt buộc |
| 4 | requestCode | String | Mã yêu cầu hoàn trả | Bắt buộc |
| 5 | amount | Int | Số tiền hoàn trả | Bắt buộc |
| 6 | orderId | String | Mã hóa đơn hoàn trả | Bắt buộc |
| 7 | orderInfo | String | Nội dung hoàn tiền | Bắt buộc |
| 8 | transactionType | String | Loại hoàn tiền  02 : hoàn tiền toàn phần | Bắt buộc |
| 9 | checksum | String | Mã đảm bảo toàn vẹn dữ liệu. Dùng giao thức Sha256 để mã hóa các thông tin gồm : secretKey +partnerCode + accessKey+ transactionNo + orderId + requestCode + amount + transactionType + orderInfo | Bắt buộc |

*Bảng 2: tham số truyền vào chuỗi json của body*

Ví dụ:

|  |
| --- |
| Chuỗi json |
| {  "partnerCode": "000.00.18.H29",  "accessKey": "f53abff08e6433696db0fdbe5e2e1d88f5f6e7624e4cd897db2ded626c0bee9b",  "transactionNo": "835651612",  "orderInfo": "Nội dung",  "requestCode": "1590278921647",  "amount": 10000,  "orderId": "050720190008",  "transactionType": "02"  } |

#### Đầu ra:

* + Chuỗi json chứa thông tin trả về , resultObject là true hoặc false

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tham số | Thuộc tính tham số | Mô tả |
| 1 | error\_code | String | Mã lỗi thực hiện yêu cầu:  SUCCESSFUL: thành công  FAILED: thất bại, NOT\_AUTHORIZED : lỗi Authorized,  ACCOUNT\_NOT\_EXIST : tài khoản không tồn tại  PARAM\_ERROR: Dữ liệu gửi lên thiếu nội dung  ORDER\_EXITS: Giao dịch đã tồn tại |
| 2 | error\_message | String | Thông tin error message |
| 3 | data | JSON | Địa chỉ trả về đường dẫn thanh toán |

*Bảng 3: tham số trả về của chuỗi json*

Ví dụ:

|  |  |
| --- | --- |
| Thành công | Lỗi |
| {  "error\_code":"SUCCESSFUL",  "error\_message":"SUCCESSFUL ",  "data":"null" | {  "error\_code": "FAILED",  "error\_message": "FAILED ",  "data": null  } |

## API tra cứu giao dịch

* Đường dẫn: /GetOrderInfo
* Method: POST

#### Đầu vào:

* + Header: Authorization
  + Body: chuỗi json chứa thông tin truyền vào

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tham số | Thuộc tính tham số | Mô tả | Lưu ý |
| 1 | partnerCode | String | partnerCode của tài khoản website dịch vụ | Bắt buộc |
| 2 | accessKey | String | Chuỗi mã hóa xác định quyền truy cập | Bắt buộc |
| 3 | transactionNo | String | Mã thanh toán | Bắt buộc |
| 4 | orderId | String | Mã hóa đơn | Bắt buộc |
| 5 | checksum | String | Mã đảm bảo toàn vẹn dữ liệu. Dùng giao thức Sha256 để mã hóa các thông tin gồm : secretKey +partnerCode + accessKey+ transactionNo + orderId |  |

*Bảng 2: tham số truyền vào chuỗi json của body*

Ví dụ:

|  |
| --- |
| Chuỗi json |
| {  "partnerCode": "000.00.18.H29",  "accessKey": "f53abff08e6433696db0fdbe5e2e1d88f5f6e7624e4cd897db2ded626c0bee9b",  "payTransId": "835651612",  "orderId": "050720190008"    } |

#### Đầu ra:

* + Chuỗi json chứa thông tin trả về , resultObject là true hoặc false

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tham số | Thuộc tính tham số | Mô tả |
| 1 | partnerCode | String | partnerCode của tài khoản website dịch vụ |
| 2 | transactionNo | String | Mã giao dịch |
| 3 | requestCode | String | Mã yêu cầu |
| 4 | amount | String | Số tiền thanh toán |
| 5 | paygate | String | Mã cổng nhận hoàn trả |
| 6 | orderId | String | Mã hóa đơn yêu cầu thanh toán |
| 7 | orderInfo | String | Nội dung thanh toán |
| 8 | payDate | String | Thời gian thanh toán |
| 9 | type | String | Loại  Pay: thanh toán  Refund: Hoàn trả |
| 10 | error\_code | String | Mã lỗi thực hiện yêu cầu :  SUCCESSFUL: thành công  FAILED: thất bại,  PARAM\_ERROR: Dữ liệu gửi lên thiếu nội dung  NOT\_EXITS: Giao dịch không tồn tại |
| 11 | error\_message | String | Message thông báo |
|  |  |  |  |

*Bảng 3: tham số trả về của chuỗi json*

Ví dụ:

|  |  |
| --- | --- |
| Thành công | Lỗi |
| {  "error\_code": "SUCCESSFUL",  "error\_message": "",  "data": {  "amount": "40000",  "partnerCode": "000.00.18.H29",  "orderId": "29.98.H29-310519-0001\_201",  "serviceCode": "hcm\_dichvucong",  "payTransId": "13333267",  "orderInfo": "Nội dung",  "errorCode": "00",  "paygate": "vnpay",  "type": "pay",  "payDate": "20200712225148"  }  } | {  "error\_code": "FAILED",  "error\_message": "",  "data": null  } |
|  |  |

# CODE EXAMPLE

## **Code mẫu .Net**

using RestSharp;

using System;

using System.Collections.Generic;

using System.Configuration;

using System.Linq;

using System.Text;

* API Method Get

public static String HTTP\_GET(String url, String urlSerivce, String sercet, Dictionary<string, string> param)

{

var client = new RestClient(url);

var request = new RestRequest(urlSerivce, Method.GET);

request.AddHeader("Authorization", sercet);

//Add Parameter vào request

foreach (var item in param)

{

request.AddParameter(item.Key, item.Value);

}

var response = client.Execute(request);

if(response.StatusCode == System.Net.HttpStatusCode.OK)

{

return response.Content;

}

else

{

//Xử lý trường hợp kết nối lỗi

return "";

}

}

* API Method POST

//jsonObj : object có cấu trúc theo như mô tả

public static String HTTP\_POST(String url, String urlSerivce, String sercet,object jsonObj)

{

var client = new RestClient(url);

var request = new RestRequest(urlSerivce, Method.GET);

request.AddHeader("Authorization", sercet);

request.AddHeader("Content-Type", "application/json;charset=UTF-8");

request.AddBody(jsonObj); // jsonObj sẽ được tự động chuyển thành chuỗi json tương ứng

var response = client.Execute(request);

if(response.StatusCode == System.Net.HttpStatusCode.OK)

{

return response.Content;

}

else

{

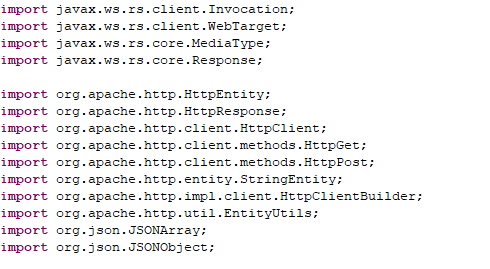
//Xử lý trường hợp kết nối lỗi

return "";

}

}

## **Code mẫu java**

****

* API Method Get

**public** **static** String HTTP\_GET(String url, **boolean** authorization, String sercet) **throws** UnsupportedEncodingException {

HttpClient httpClient = HttpClientBuilder.*create*().build();

HttpGet httpGet = **new** HttpGet(url);

**if** (authorization) {

httpGet.setHeader("Authorization", sercet);

}

String responseString = "";

**try** {

HttpResponse response = httpClient.execute(httpGet);

HttpEntity entity = response.getEntity();

**if** (entity != **null**) {

responseString = EntityUtils.*toString*(entity, "UTF-8");

}

} **catch** (IOException e) {

e.printStackTrace();

}

System.***out***.println(responseString);

**return** responseString;

}

* API Method POST

**public** **static** String HTTP\_POST(String url, **boolean** authorization, String sercet, String dataJson) **throws** UnsupportedEncodingException {

HttpClient httpClient = HttpClientBuilder.*create*().build();

HttpPost httpPost = **new** HttpPost(url);

**if** (authorization) {

httpPost.setHeader("Authorization", sercet);

}

httpPost.setHeader("Content-Type", "application/json;charset=UTF-8");

StringEntity httpEntity = **new** StringEntity(dataJson, "UTF-8");

httpPost.setEntity(httpEntity);

String responseString = "";

**try** {

HttpResponse response = httpClient.execute(httpPost);

HttpEntity entity = response.getEntity();

**if** (entity != **null**) {

responseString = EntityUtils.*toString*(entity, "UTF-8");

}

} **catch** (IOException e) {

e.printStackTrace();

}

System.***out***.println(responseString);

**return** responseString;

}

# Tham khảo

* Thông tin tài liệu được cập nhật tại địa chỉ: <https://api.tphcm.gov.vn/ChiTietLgsp/id/65/idPa/68/mota/MoTa>