

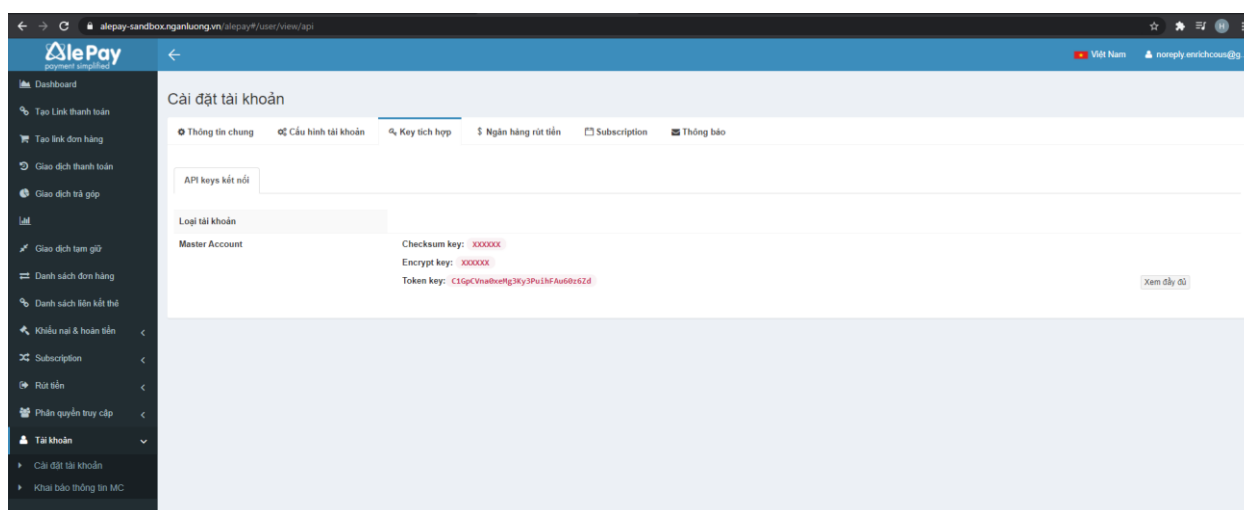
# Thanh toán Alepay IMS VIỆT NAM

## 1. Cấu hình:

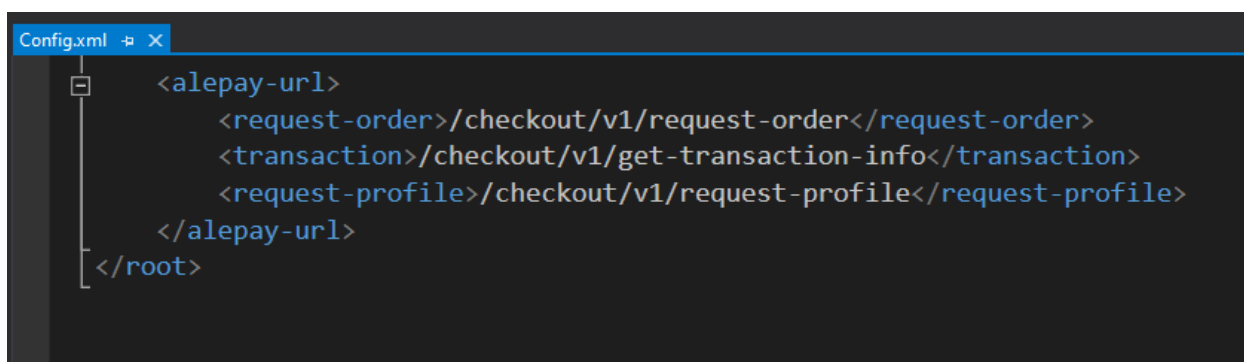
### a. Cấu hình trong file web.config:

```
<!--Payment Alepay-->
<add key="TokenKey_Alepay" value="C1GpCVna0xeMg3Ky3PuihFAu60z6Zd" />
<add key="ChecksumKey_Alepay" value="HHDfbWnQpT1AaF9HMkoGN1JdLBgxKe" />
<add key="EncryptKey_Alepay" value="MIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQC1pnljDY2of7UuUQdRME0RepCK8UV9IzZVnKcugTO7Vxv
<add key="APIUrl_Alepay" value="https://alepay-sandbox.nganluong.vn"/>
```

Các key được Alepay cung cấp trong account đã đăng ký ở url APIUrl\_Alepay.



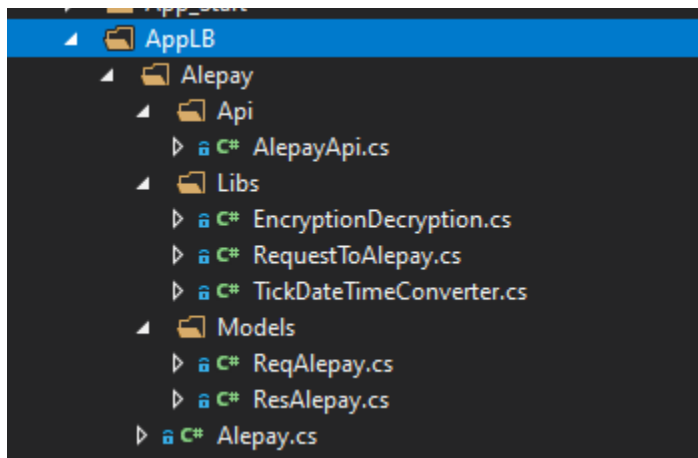
### b. Cấu hình trong file App\_Data/Config.xml:



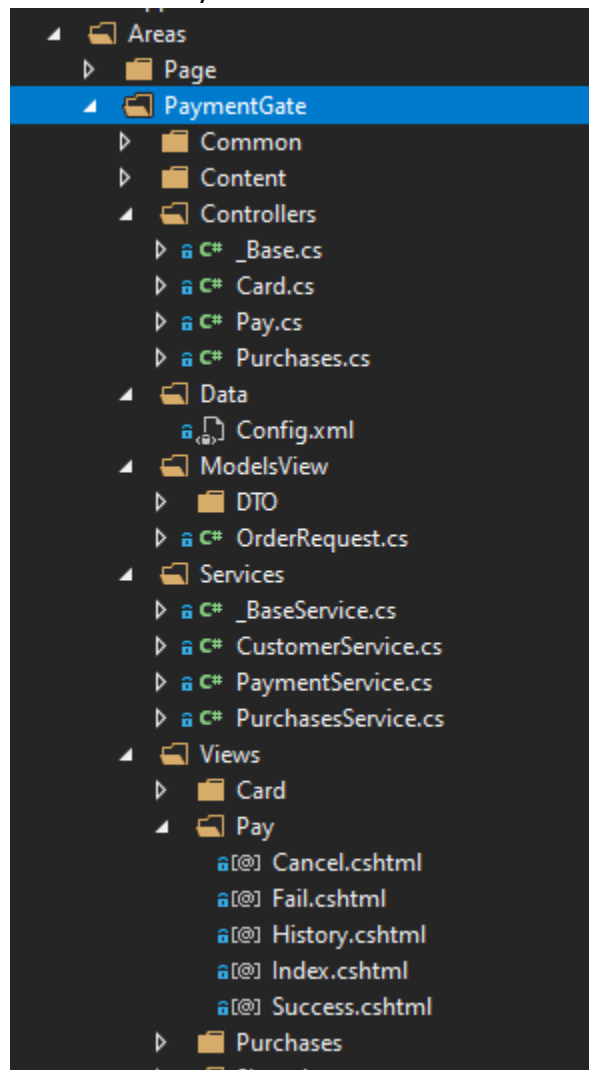
Đường link trở đến các api của Alepay

## 2. Cấu trúc thư mục:

### a. Xây dựng thư viện Alepay



### b. Controller xử lý thanh toán



### 3. Chức năng các function, flow xử lý:

#### a. Thư viện Alepay:

##### i. Libs:

##### 1. Encryption, Decryption code

```
1  using ...
15
16  namespace EnrichcousBackOffice.AppLB.Alepay.Libs
17  {
18      8 references
19      public class EncryptionDecryption
20      {
21          static RsaKeyParameters rsaKeyParameters;
22          static string publicKey = ConfigurationManager.AppSettings["EncryptKey_Alepay"];
23          0 references
24          static EncryptionDecryption()...
25
26          1 reference
27          public static string RSAEncrypt(object obj)...
```

##### 2. RequestToAlepay: xử lý request và response api đến hệ thống Alepay

```
RequestToAlepay.cs
1  using ...
12
13  namespace EnrichcousBackOffice.AppLB.Alepay.Libs
14  {
15      3 references
16      public class RequestToAlepay
17      {
18          private static string token = ConfigurationManager.AppSettings["TokenKey_Alepay"];
19          private static string checksumKey = ConfigurationManager.AppSettings["ChecksumKey_Alepay"];
20          3 references
21          public static string sendRequest(object obj, string url)...
```

a. Data request Alepay:

Request data		
Trường	Kiểu dữ liệu	Mô tả dữ liệu
token	String	Token Key Alepay đã cung cấp
data	String	Data đã được mã hóa (Sử dụng Encrypt key Alepay cung cấp) <ul style="list-style-type: none"> <li>Cấu trúc tạo data mã hóa : <ul style="list-style-type: none"> <li>Encode data sang dạng Json</li> <li>Thực hiện mã hóa chuỗi JSON sử dụng thuật toán RSA với public key đã lấy từ Alepay.</li> </ul> </li> </ul>
checksum	String	Check sum cho data đã được mã hóa (Sử dụng Checksum key Alepay cung cấp) <ul style="list-style-type: none"> <li>Cấu trúc tạo Checksum : <ul style="list-style-type: none"> <li>Thực hiện lấy chuỗi data đã mã hóa + key checksum và thực hiện md5 : Checksum = MD5(dataEncrypted + ChecksumKey)</li> </ul> </li> </ul>

b. Data response:

Response data		
Trường	Kiểu dữ liệu	Mô tả dữ liệu
errorCode	String	Mã lỗi ( được mô tả trong Phụ lục V.1 )
data	String	Data đã được mã hóa (Sử dụng Encrypt key Alepay cung cấp)
checksum	String	Check sum cho data đã được mã hóa (Sử dụng Checksum key Alepay cung cấp)
errorDescription	String	Mô tả lỗi

## ii. Api

### 1. Request các api đến alepay bằng Libs/RequestToAlepay

```
1  using ...
12
13  namespace EnrichcousBackOffice.AppLB.Alepay.Api
14  {
15      2 references
16      public class AlepayApi
17      {
18          1 reference
19          public static string RequestOrder(Order reqData)
20          {
21              var url = ConfigurationManager.AppSettings["APIUrl_Alepay"] + AppConfig.Cfg.AlepayUrl.RequestOrder;
22              var resultJson = RequestToAlepay.sendRequest(reqData, url);
23              ResAlepay response_data = JsonConvert.DeserializeObject<ResAlepay>(resultJson);
24
25              ResAlepayView resView = new ResAlepayView { };
26              var settings = new JsonSerializerSettings()
27              {
28                  ContractResolver = new CamelCasePropertyNamesContractResolver(),
29                  Converters = new List<JsonConverter> { new TickDateTimeConverter() }
30              };
31
32              if (response_data.errorCode == "000")
33              {
34                  resView.error = false;
35                  resView.data = EncryptionDecryption.RSADecrypt<ResCheckout>(response_data.data);
36              }
37              else
38              {
39                  resView.error = true;
40                  resView.errorDescription = response_data.errorDescription;
41              }
42              return JsonConvert.SerializeObject(resView, settings);
43          }
44
45          1 reference
46          public static string RequestTransaction(string transaction) ...
47
48          0 references
49          public static string RequestProfile(ProfileAlepay reqData) ...
50      }
51  }
```

- a. Input: data request api.
- b. Output: data response api.
- c. Process: Libs/ResquestToAlepay/sendRequest()

### 2. Chức năng từng function:

- a. RequestOrder(): request đến Alepay để trả về 1 link thanh toán khách hàng nhập thông tin thẻ vào để thanh toán.
- b. RequestTransaction(): request Alepay trả về thông tin 1 transaction thanh toán bằng transaction code.
- c. RequestProfile(): request Alepay trả về thông tin thẻ khách hàng đã liên kết thẻ.

### iii. Alepay.cs

```
Alepay.cs
EnrichcousBackOffice
EnrichcousBackOffice.AppLB.Alepay
returnUrl

1  using ...
2
3
4
5
6
7
8  namespace EnrichcousBackOffice.AppLB.Alepay
9  {
10     4 references
11     public class Alepay
12     {
13         private static string returnUrl = HttpContext.Current.Request.Url.GetLeftPart(UriPartial.Authority) + "/paymentgate/pay/processing";
14         private static string cancelUrl = HttpContext.Current.Request.Url.GetLeftPart(UriPartial.Authority) + "/paymentgate/pay/cancel?key=";
15         2 references
16         public static string GetInforPayment(string transaction)...
17         1 reference
18         public static string CheckoutPayment(object reqOrder)
19         {
20             Order req = JsonConvert.DeserializeObject<Order>(JsonConvert.SerializeObject(reqOrder));
21             var orderInfo = new Order...;
22             47
23             string result = AlepayApi.RequestOrder(orderInfo);
24             49
25             return result;
26         }
27         1 reference
28         public static T PaymentSuccess<T>(string data, string checksum)...
29     }
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
```

1. CheckoutPayment():
  - a. Input: thông tin của merchant.
  - b. Output:
    - i. Message thanh toán
    - ii. Link thanh toán từ Alepay
  - c. Process: Api/AlepayApi/RequestOrder
2. PaymentSuccess():
  - a. Sau khi thực hiện thanh toán từ Link thanh toán từ Alepay Alepay sẽ return về link return mà đã set trong thông tin merchant . Url get trả về chứa data và checksum.
  - b. Input: data và checksum code từ mục a.
  - c. Output: json data transaction code.
  - d. Process: RSADecrypt.

b. Controller xử lý thanh toán:

```
Pay.cs  PayController : Base
{
    2 references
    internal CardUtil cardUtil = new CardUtil();
    internal CustomerService _customer = new CustomerService();
    internal PaymentService _payment = new PaymentService();

    private WebDataModel db = new WebDataModel();

    /// <summary> Login
    0 references
    public JsonResult Login(string email, string password)...

    2 references
    public string GenerateLinkPayment(string invoice)...

    0 references
    public ActionResult Index(string invoice)...

    0 references
    public JsonResult RequestPayment(OrderRequest req)...

    0 references
    public async Task<ActionResult> Processing(string data, string checksum)...

    0 references
    public ActionResult Success()...

    0 references
    public ActionResult Fail()...

    0 references
    public ActionResult Cancel(string key)...

    0 references
    public ActionResult History(string key)...
}
```

i. RequestPayment()

1. Input: thông tin đơn hàng, khách hàng
2. Output: thông tin từ Alepay/PaymentSuccess()
3. Process: call function Alepay/ CheckoutPayment()

ii. Processing()

1. Đây là url được Alepay return về để xử lý data, checksum code, từ data đó có thể lưu transaction cho lần thanh toán này.
2. Input: data, checksum do Alepay return
3. Output: trả về trạng thái của lần thanh toán này (Success, Fail)
4. Process: Lưu data transaction vào DB.

iii. Success / Fail / Cancele

1. Status của lượt thanh toán.