

Tài liệu hướng dẫn tích hợp App/Game và Zing Credits dùng phương pháp Push

(Document version 1.0 - Vietnamese)

Tác giả Nguyễn Quốc Toàn

Liên hệ toannq@vng.com.vn

Tác giả Nguyễn Quang Nam

Liên hệ namnq@vng.com.vn

MỤC LỤC

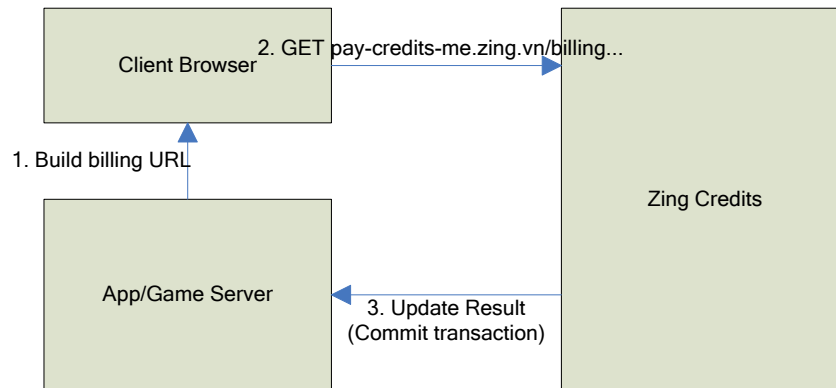
| | |
|--|-----------|
| MỤC LỤC..... | 2 |
| 1 - Giới thiệu | 3 |
| 2 - Tổng quan..... | 4 |
| Mô hình hoạt động của Zing Credits..... | 4 |
| UI flow | 5 |
| 3 - To do | 6 |
| Task flow..... | 6 |
| 3.1 - Đăng ký..... | 6 |
| 3.2 - Đồng bộ thời gian | 7 |
| 4 - Mã hóa dữ liệu | 8 |
| 4.1 - Hàm mã hóa..... | 8 |
| 4.2 - Hàm giải mã..... | 9 |
| 5 - Billing..... | 11 |
| Billing flow | 11 |
| 5.1 - Thông tin mã hóa | 12 |
| 5.2 - Tra cứu số dư | 13 |
| 6 - Update Result (Commit transaction)..... | 14 |
| 6.1 - Thông tin giải mã..... | 14 |
| 6.2 - REST respond và mã lỗi | 15 |

1 - Giới thiệu

Zing Credits là một core-application trên Zing Me hỗ trợ cho việc thanh toán các vật phẩm hoặc dịch vụ trong App/Game, hoặc chuyển tiền vào ví riêng của App/Game. Đơn vị tiền trong Zing Credits là Zing Xu, là đơn vị được dùng chung cho các sản phẩm của công ty VNG. Zing Credits hiện là kênh thanh toán duy nhất cho các App/Game ngoài VNG.

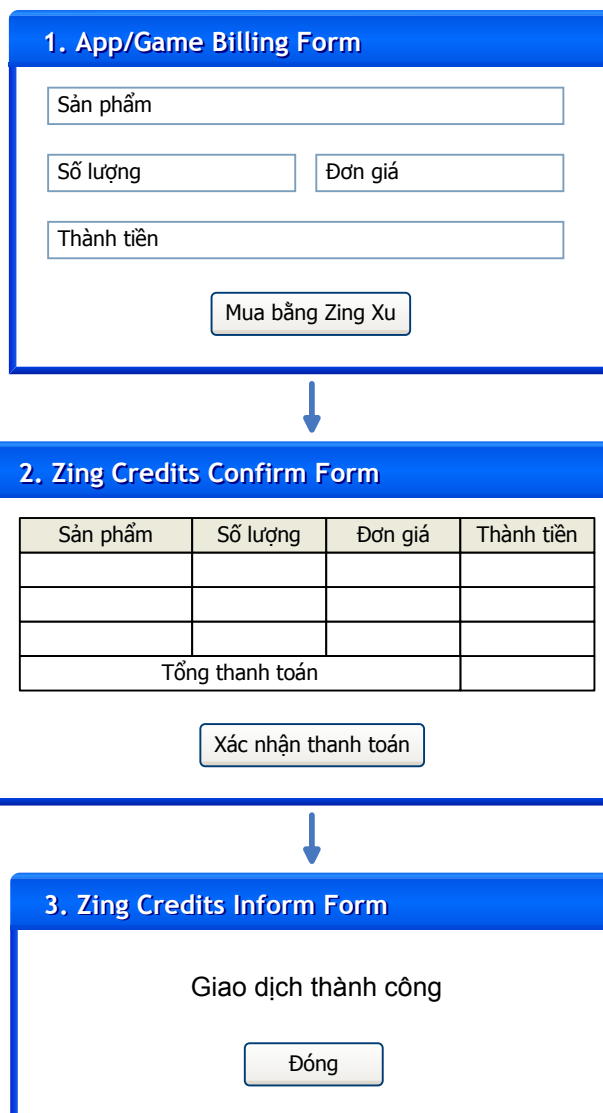
Zing Credits hiện hỗ trợ thanh toán theo phương pháp Push (tức là App/Game gửi yêu cầu thanh toán đến Zing Credits, Zing Credits cập nhật tài khoản của người dung, sau đó gửi lại một yêu cầu cập nhật kết quả cho App/Game). Giao thức được dùng trong API của Zing Credits là REST over HTTP

2 - Tổng quan



Mô hình hoạt động của Zing Credits

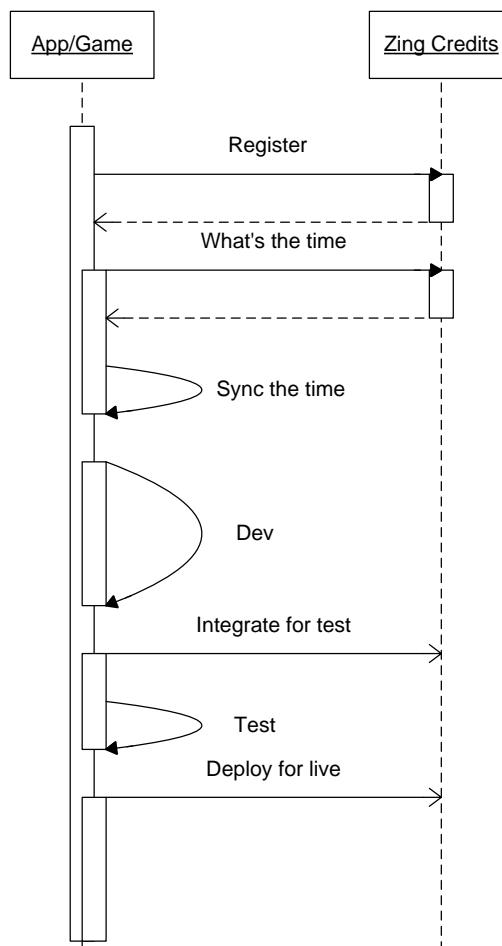
- 1) Khi user yêu cầu mua vật phẩm bằng Zing Xu, App/Game Server sẽ tạo ra billing URL, URL này được xây dựng dựa trên giải thuật mã hóa do Zing Credits đưa ra. Sau khi có billing URL App/Game Server trả lại cho Client Browser để từ đó submit lên Zing Credits
- 2) Từ Client Browser, user được redirect đến billing URL (GET method). Zing Credits sẽ verify URL này và trả lại cho user 1 form để xác nhận thông tin lần nữa. Nếu user đồng ý, Zing Credits sẽ xử lý yêu cầu thanh toán (trừ Zing Xu trong tài khoản)
- 3) Zing Credits gọi callback function “UpdateResult” cho App/Game (REST over http) sau khi đã cập nhật tài khoản của user. Tại bước này App/Game chính thức chuyển vật phẩm vào tài khoản app/game cho user.



UI flow

- 1) Billing Form: Form được tạo ra bởi App/Game sau khi có yêu cầu thanh toán bằng Zing Xu
- 2) Confirm Form: Form tạo ra bởi Zing Credits sau khi user được App/Game redirect tới billing URL
- 3) Inform Form: Form tạo ra bởi Zing Credits sau khi Zing Credits xử lý giao dịch thành công

3 - To do



Task flow

3.1 - Đăng ký

Bên App/Game cần cung cấp cho bên Zing Credits các thông tin sau:

- Tên của App/Game: hiển thị trên UI, độ dài dưới 256 kí tự. Ví dụ “*Siêu thị BẠN BÈ*”
- Mô tả: đoạn văn bản mô tả về app/game, độ dài dưới 1024 kí tự. Ví dụ “*Trò chơi mua bạn bè làm thú cưng*”
- URL: đường dẫn URL trên Zing Me của app/game
- Icon: file ảnh đại diện cho app/game, kích thước 64x64 pixels

- UpdateResult URL: đường dẫn URL để Zing Credits gửi request UpdateResult (commit transaction) sau khi đã xử lý yêu cầu thanh toán (cập nhật trừ Zing Xu trong tài khoản). Ví dụ: “*http://ffs.apps.zing.vn/app/payment/updateresult*”

Sau khi đăng ký thông tin App/Game vào hệ thống, Zing Credits gửi lại cho App/Game các thông tin sau:

- AppID: ID đại diện cho app/game dùng trong tất cả quá trình trao đổi dữ liệu với Zing Credits
- Key1: Mã bí mật dùng cho việc mã hóa dữ liệu khi tạo billing URL
- Key2: Mã bí mật dùng cho việc giải mã dữ liệu khi xử lý yêu cầu UpdateResult từ Zing Credits
- Auth Method: phương pháp thực hiện mã hóa/giải mã dữ liệu (là tham số cho hàm mã hóa/giải mã)

Lưu ý: Bên App/Game có trách nhiệm giữ bí mật Key1, Key2 sau khi nhận được.

3.2 - Đồng bộ thời gian

Mỗi yêu cầu thanh toán chỉ có hiệu lực trong khoảng thời gian **5 phút**. Quá khoảng thời gian này hệ thống Zing Credits sẽ từ chối xử lý. Thời gian của mỗi yêu cầu được tính từ khi App/Game tạo ra billing URL đến khi Zing Credits nhận được billing URL. Hệ thống Credits chấp nhận sai số về thời gian do tương tác của user và độ trễ trên network là **5 phút**. Như vậy tổng thời gian chênh lệch không được phép quá 10 phút. Do đó Server App/Game có nhiệm vụ build billing URL và Server của Zing Credits cần được đồng bộ thời gian sao cho không lệch quá **10 phút**.

Phương pháp đồng bộ thời gian như sau:

- Sử dụng thời gian chuẩn từ một Time Server độc lập khác mà cả 2 bên App/Game & Credits đều có thể truy xuất được.
- Bên Zing Credits sẽ gửi cho bên App/Game độ lệch thời gian với Time Server trên để bên App/Game điều chỉnh thời gian hệ thống của Server cho phù hợp.

Thông tin về thời gian hệ thống của Server Credits sẽ được gửi kèm ngay sau khi App/Game thực hiện đăng ký với hệ thống Zing Credits.

4 - Mã hóa dữ liệu

Việc mã hóa/giải mã dữ liệu được thực hiện nhờ thư viện mã hóa **libZCypher** do Zing Credits cung cấp. Hiện nay Zing Credits hỗ trợ thư viện cho các ngôn ngữ lập trình C#, Java, C++, PHP.

4.1 - Hàm mã hóa

```
int zma_encode(zcommon_StringHolder strData_o
, std_vectorT_std_string_t arStrData_i
, zcommon_StringHolder strKey_i
, int iMethod_i
, long uiReserve_i);
```

Trong đó:

- Return: Mã lỗi trong việc mã hóa. ≥ 0 nếu thành công, < 0 nếu thất bại.
- Output param *strData_o*: Dữ liệu dạng chuỗi sau khi mã hóa
- Input param *arStrData_i*: Mảng các chuỗi dữ liệu cần mã hóa, cụ thể trong quá trình billing là các thông tin của một yêu cầu thanh toán.
- Input param *strKey_i*: Key bí mật **Key1** (được cung cấp sau khi đăng ký App/Game)
- Input param *iMethod_i*: Phương pháp mã hóa (được cung cấp sau khi đăng ký App/Game)
- Input param *uiReserve_i*: Không dùng. Yêu cầu giá trị là 0.

Ví dụ:

```
//setup params
std_vectorT_std_string_t params = new std_vectorT_std_string_t();
params.add("user ID");
params.add("bill no");
params.add("item IDs");
params.add("item names");
params.add("1"); //quantities
params.add("100"); //prices
params.add("100"); //amount
params.add("123456780"); //unix timestamp
//setup key
zcommon_StringHolder key = new zcommon_StringHolder();
key.setValue("private key of app/game");
//encode
zcommon_StringHolder encoded_data = new zcommon_StringHolder();
int e = ZCypher.zma_encode(encoded_data, params, key, 0, 0);
```



```
//export result
if (e >= 0) {
    System.out.println("Encoded data: " + encoded_data.getValue());
} else {
    System.out.println("Error: " + e);
}
```

4.2 - Hàm giải mã

```
int zma_decode(std__vectorT_std__string_t arStrData_o
, zcommon_StringHolder strData_i
, zcommon_StringHolder strKey_i
, int iMethod_i
, long reserve);
```

Trong đó:

- Return: Mã lỗi trong việc giải mã. ≥ 0 nếu thành công, < 0 nếu thất bại.
- Output param *arStrData_o*: Mảng các chuỗi dữ liệu sau khi giải mã, cụ thể khi xử lý ReturnResult là các thông tin của một yêu cầu thanh toán trước đó.
- Input param *strData_i*: Dữ liệu dạng chuỗi mã hóa cần được giải mã
- Input param *strKey_i*: Key bí mật **Key2** (được cung cấp sau khi đăng ký App/Game)
- Input param *iMethod_i*: Phương pháp mã hóa (được cung cấp sau khi đăng ký App/Game)
- Input param *reserve*: Chưa dùng. Yêu cầu giá trị là 0.

Ví dụ:

```
//decode
std__vectorT_std__string_t params = new std__vectorT_std__string_t();
e = ZCypher.zma_decode(params, encoded_data, key, 0, 0);
//export result
if (e >= 0) {
    if (params.size() >= 8) {
        System.out.println("user ID =" + params.get(0));
        System.out.println("bill no =" + params.get(1));
        System.out.println("item IDs =" + params.get(2));
        System.out.println("item names =" + params.get(3));
        System.out.println("quantities =" + params.get(4));
        System.out.println("prices =" + params.get(5));
        System.out.println("amount =" + params.get(6));
        System.out.println("timestamp =" + params.get(7));
    }
}
```

App/Game & Zing Credits Integration Document

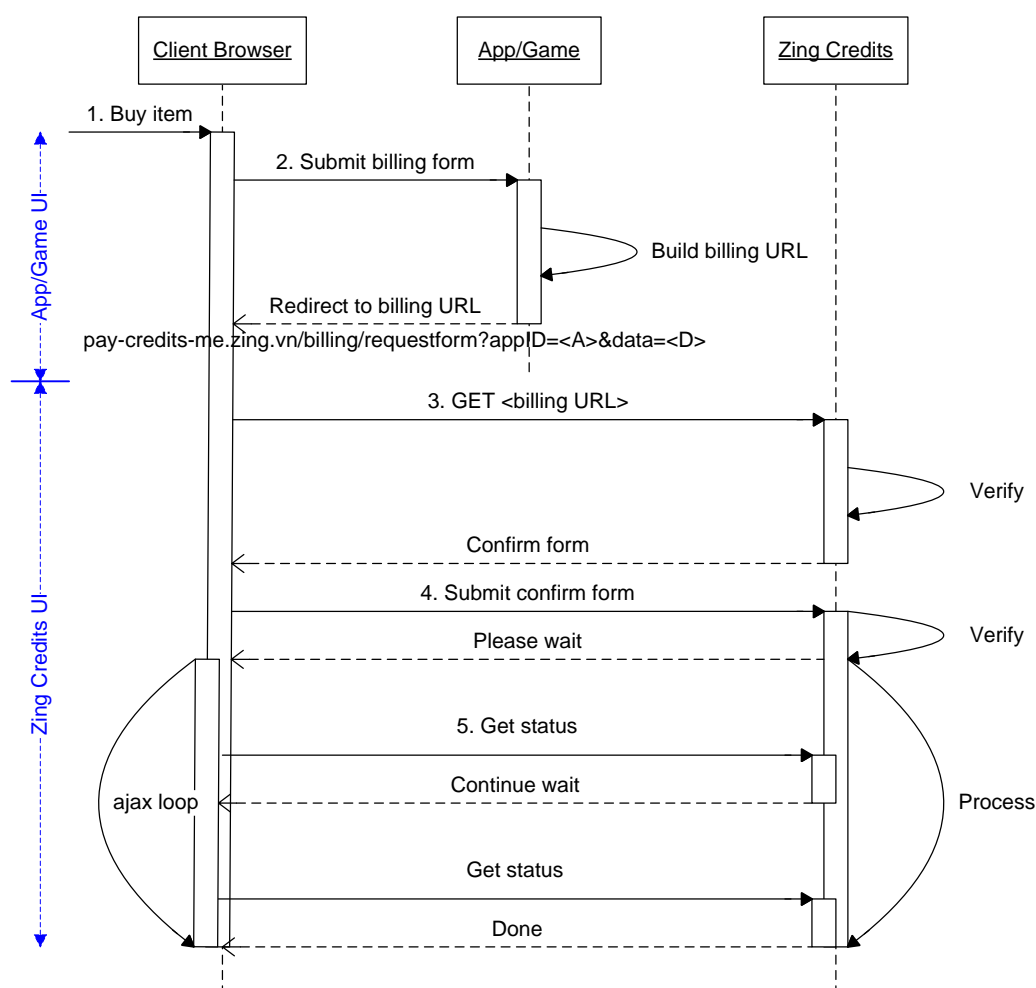
```
    } else {  
        System.out.println("Decoded successfully, but the size is  
wrong!");  
    }  
} else {  
    System.out.println("Error: " + e);  
}
```

5 - Billing

Zing Credits mở API cho billing dưới dạng HTTP request. Billing URL có định dạng như sau: <http://pay-credits-me.zing.vn/billing/requestform?appID=<A>&data=<D>>

Trong đó <A> là ID của App/Game được bên Zing Credits gửi cho bên App/Game sau khi đăng ký App/Game vào hệ thống Credits. <D> là chuỗi dữ liệu sau khi mã hóa tất cả các thông tin cần thiết của yêu cầu thanh toán.

Cách gửi yêu cầu thanh toán do phía App/Game quyết định. Dưới đây là flow mà bên Zing Credits đề nghị



Billing flow

Bước 1. Buy Item & 2. Submit billing form thực thi từ App/Game site. Khi nhận được submitted billing form, App/Game server cần build billing URL dung giải thuật mã hóa đã nêu trong mục trước để trả lại cho user side (redirect - 302).

Bước 3. *GET <billing URL>* Tại user side sau khi có được billing URL cần gửi một GET request với billing URL đó đến server Zing Credits. Bước này nên được thực hiện trên một normal window mới hoặc là một popup window.

Bước 4. *Submit confirm form* & 5. *Get status* thực thi tại Zing Credits site.

5.1 - Thông tin mã hóa

Tham số data trong Billing URL được xây dựng bằng cách mã hóa (hàm `zma_encode` trong `libZCypher`) các thông tin sau:

| Index | Nội dung | Kiểu dữ liệu | Giải thích |
|-------|----------------|------------------------|---|
| 0 | userID | Int32 | Zing Me account ID của user |
| 1 | billNo_AppGame | String | ID của giao dịch thanh toán, tồn tại duy nhất cho mỗi giao dịch. billNo được tạo ra bởi App/Game, và được dùng trong việc đối soát số liệu giữa App/Game và Zing Credits |
| 2 | itemIDs | String | ID của sản phẩm trong giao dịch thanh toán. ID này do App/Game quản lý |
| 3 | itemNames | String (UTF8) | Tên (printable) của sản phẩm trong giao dịch thanh toán. Tên này do App/Game quản lý, và được hiển thị trong trang nhật ký giao dịch của user trên Zing Credits |
| 4 | itemQuantities | Int32 | Số lượng sản phẩm trong giao dịch thanh toán |
| 5 | itemPrices | Double | Giá (Zing Xu) của sản phẩm trong giao dịch thanh toán. Giá cho phép lẻ đến 1% của 1 Zing Xu, ví dụ 0.01 |
| 6 | amount | Double | Tổng số Zing Xu cần thanh toán trong giao dịch, cho phép giá trị lẻ đến 1% của 1 Zing Xu. Giá trị này sẽ được trừ đi trong tài khoản Zing Credits của user. Thường là amount = itemQuantities x itemPrices |
| 7 | timestamp | Int32 (Unix timestamp) | Thời điểm tạo giao dịch thanh toán. Đây là local time (local timezone) của App/Game Server, và được dùng trong việc đối soát số liệu giữa App/Game và Zing Credits |

Lưu ý:

- Tên field dữ liệu (userID, billNo, ...) chỉ dùng để giải thích, không được bao gồm vào các tham số cho hàm mã hóa
- Các kiểu dữ liệu khác nhau đều cần được chuyển đổi thành kiểu String khi dùng hàm mã hóa
- Thứ tự các tham số cần đúng như trong bảng khi truyền vào hàm mã hóa

5.2 - Tra cứu số dư

App/Game có thể có nhu cầu kiểm tra số dư của user trước khi gửi yêu cầu thanh toán hoặc hiển thị thông tin về số dư của user trên UI. Tương tự như Billing, Zing Credits cung cấp API cho balance query dưới dạng HTTP request. Balance URL có định dạng như sau: <http://pay-credits-me.zing.vn/billing/balance?appID=<A>&data=<D>>

Trong đó <A> là ID của App/Game, <D> là chuỗi dữ liệu sau khi mã hóa các thông tin cần thiết của yêu cầu truy vấn.

Tham số data trong Balance URL được xây dựng bằng cách mã hóa (hàm zma_encode trong libZCypher) các thông tin sau:

| Index | Nội dung | Kiểu dữ liệu | Giải thích |
|-------|----------|--------------|-----------------------------|
| 0 | userID | Int32 | Zing Me account ID của user |

Dữ liệu trả về dưới dạng REST respond: dòng đầu tiên là số dư tài khoản. App/Game cần parse String để chuyển đổi thành kiểu số Double

6 - Update Result (Commit transaction)

Sau khi cập nhật tài khoản của user trong giao dịch thanh toán (trừ Zing Xu), Zing Credits sẽ gửi 1 thông báo cho bên App/Game để App/Game chuyển sản phẩm đến user. Thông báo này là 1 HTTP request có định dạng URL như sau:

<UpdateResult Path>?data=<D>

Trong đó **<UpdateResult Path>** là đường dẫn xử lý UpdateResult do App/Game cung cấp khi đăng ký; **<D>** là chuỗi dữ liệu sau khi mã hóa tất cả các thông tin cần thiết của yêu cầu thanh toán, chi tiết các thông tin được mã hóa trong **<D>** được nêu rõ trong mục bên dưới

6.1 - Thông tin giải mã

App/Game khi xử lý 1 UpdateResult request, cần giải mã (hàm zma_decode trong libZCypher) chuỗi data trong URL. Dữ liệu sau khi được giải mã chứa các thông tin sau:

| Index | Nội dung | Kiểu dữ liệu | Giải thích |
|-------|----------------|------------------------|---|
| 0 | userID | Int32 | Thông tin từ yêu cầu thanh toán (Tham khảo mục: Billing – Thông tin mã hóa) |
| 1 | billNo | String | Thông tin từ yêu cầu thanh toán (Tham khảo mục: Billing – Thông tin mã hóa) |
| 2 | itemIDs | String | Thông tin từ yêu cầu thanh toán (Tham khảo mục: Billing – Thông tin mã hóa) |
| 3 | itemNames | String (UTF8) | Thông tin từ yêu cầu thanh toán (Tham khảo mục: Billing – Thông tin mã hóa) |
| 4 | itemQuantities | Int32 | Thông tin từ yêu cầu thanh toán (Tham khảo mục: Billing – Thông tin mã hóa) |
| 5 | itemPrices | Double | Thông tin từ yêu cầu thanh toán (Tham khảo mục: Billing – Thông tin mã hóa) |
| 6 | amount | Double | Thông tin từ yêu cầu thanh toán (Tham khảo mục: Billing – Thông tin mã hóa) |
| 7 | timestamp | Int32 (Unix timestamp) | Thông tin từ yêu cầu thanh toán (Tham khảo mục: Billing – Thông tin mã hóa) |

| | | | |
|---|------------------|--------|--|
| 8 | txID_ZingCredits | String | ID của giao dịch thanh toán, được tạo ra và quản lý bởi Zing Credits |
|---|------------------|--------|--|

6.2 - REST respond và mã lỗi

Kết quả trả về cho Zing Credits sau App/Game xử lý UpdateResult có định dạng REST respond như sau:

- Trên 1 dòng
- Gồm 2 phần result_code và error_message ngăn cách nhau bởi dấu 2 chấm ':'

| | | |
|---------------|---------------|---|
| result_code | Int32 | Mã lỗi được định nghĩa bởi App/Game. <ul style="list-style-type: none"> - Mã thành công (Sản phẩm được chuyển đến user): ≥ 1000 - Mã thất bại (Sản phẩm không được chuyển đến user): < -1000 |
| error_message | String (UTF8) | Chuỗi mô tả lỗi trong trường hợp result_code là thất bại, trường hợp thành công có thể để rỗng (respond vẫn có dấu 2 chấm ':') |

Ví dụ: "-1001:Sản phẩm 'Chó quý tộc' không được bán nữa"

Hoặc "1000:Thành công"

Hoặc "1001:"

Hành xử của Zing Credits khi nhận được result_code là thất bại: Zing Credits sẽ trả lại Zing Xu cho user (cộng Zing Xu bằng với lượng đã trừ trước đó - amount), và đóng giao dịch

Khi Zing Credits nhận được **result_code** là thành công sẽ đóng giao dịch.

Lưu ý: Trường hợp ngoại lệ (network chậm hoặc chập chờn), Zing Credits không nhận được **result_code** từ App/Game, Zing Credits sẽ re-try UpdateResult request 1 số lần nhất định. Nên App/Game cần kiểm tra trùng billNo trước khi thực hiện chuyển sản phẩm cho user.