**Web API**

**CATEGORY**

* Steam
* Origin
* Uplay
* Ubisoft
* Gog
* Blizzard

**Final Class CODE**

CONST String SUCCESS = 200

CONST String DATA\_INVALID = 401

CONST String DATA\_DUPLICATED= 402

CONST String DATA\_NOTFOUND = 404

**Final Class MESSAGE**

CONST String SUCCESS = “Success”

CONST String DATA\_INVALID \_USERNAME = “User name is invalid”

CONST String DATA\_INVALID \_PASSWORD= “Password is invalid”

CONST String DATA\_INVALID \_PHONE= “Phone is invalid”

CONST String DATA\_INVALID \_PHONE= “Email is invalid”

CONST String DATA\_DUPLICATED\_USERNAME = “Username has been used”

CONST String DATA\_DUPLICATED\_PHONE= “Phone has been used”

CONST String DATA\_NOTFOUND\_USERNAME = “Username is not found”

CONST String DATA\_NOTFOUND\_PASSWORD= “Password is incorrect”

CONST String DATA\_NOTFOUND\_USERID= “User ID not found”

CONST String DATA\_NOTFOUND\_CATEGORY= “Category not found”

CONST String DATA\_NOTFOUND\_PRODUCT= “ProductID not found”

CONST String DATA\_NOTFOUND\_PRODUCTINCART= “Product is not exit in cart”

**Response form**

String code

String message

String content (Json)

1. **Register and Login**
2. **Create-new-account (POST) ( /api-create-user)**
   1. Request:

* Request Body:

{

“account”: “quanghung99”,

“password”: “123456789”,

“fullname” : “Nguyen Quang Hung”,

“phone” : “0868724828”,

“email” : “email”

“address”: “Gia lam – Ha Noi”

}

* 1. Response:
* Success:

{

“code” : CODE.SUCCESS,

“message” : MESSAGE.SUCCESS,

“content” : null

}

* Username invalid:

{

“code” : CODE.DATA\_INVALID,

“message” : MESSAGE.DATA\_INVALID\_USERNAME,

“content” : null

}

* Username has been used

{

“code” : CODE.DATA\_DUPLICATED,

“message” : MESSAGE.DATA\_DUPLICATED\_USERNAME,

“content” : null

}

1. **Login (POST) (/api-login-user)**
   1. Request

* Request Body

{

“username”: “quanghung99”,

“password”: “123456789”

}

* 1. Response:
* Success

{

“code” : CODE.SUCCESS,

“message” : MESSAGE.SUCCESS,

“userID”: “idabc123”

}

* Username not found

{

“code” : CODE.DATA\_NOTFOUND,

“message” : MESSAGE.DATA\_NOTFOUND\_USERNAME,

“content” : null

}

1. **Get user info (POST) (/api-get-user-info)**
   1. Request

* Request Body

{

“userID”: “idabc123”

}

* 1. Response
* Success

{

“code” : CODE.SUCCESS,

“message” : MESSAGE.SUCCESS,

“userID”: “idabc123”

“name”: “Nguyen Quang Hung”

}

* UserID not found

{

“code” : CODE.DATA\_NOTFOUND,

“message” : MESSAGE.DATA\_NOTFOUND\_USERID

“content” : null

}

1. **Products**
2. **Get products by category (POST) (/api-find-product-by-category)**
   1. Request:

* Request Body

{

“category” : “steam”,

}

* 1. Response
* Success

{

“code” : CODE.SUCCESS,

“message” : MESSAGE.SUCCESS,

“listProduct”: [

{

“id” : “xyz456”,

“name”: “PUBG”,

“price” : 150000,

“des” : “Game of the year”,

“image”: “456”,

“category”: “steam”

},

{

“id” : “xyz457”,

“name”: “CSGO”,

“price” : 160000,

“des” : “Game of the year”,

“image”: “457”,

“category”: “steam”

}

]

}

* Category not found

{

“code” : CODE.DATA\_NOTFOUND,

“message” : MESSAGE.DATA\_NOTFOUND\_CATEGORY

“content” : null

}

1. **Get product by name (POST) // search**
   1. Request

* Request body

{

“keyword”: “pubg”

}

* 1. Response
* Success:

{

“code” : CODE.SUCCESS,

“message” : MESSAGE.SUCCESS,

“listProduct”: [

{

“id” : “xyz456”,

“name”: “PUBG”,

“price” : 150000,

“des” : “Game of the year”,

“image”: “456”,

“category”: “steam”

},

{

“id” : “xyz457”,

“name”: “CSGO”,

“price” : 160000,

“des” : “Game of the year”,

“image”: “457”,

“category”: “steam”

}

]

}

* Không có product nào match thì content null , code vẫn success

1. **Get all product (GET) (/api-get-all-product)**
   1. Request:
   2. Response:

* Success:

{

“code” : CODE.SUCCESS,

“message” : MESSAGE.SUCCESS,

“listProduct”: [

{

“id” : “xyz456”,

“name”: “PUBG”,

“price” : 150000,

“des” : “Game of the year”,

“image”: “456”,

“category”: “steam”

},

{

“id” : “xyz457”,

“name”: “CSGO”,

“price” : 160000,

“des” : “Game of the year”,

“image”: “457”,

“category”: “steam”

}

]

}

1. **Cart**
2. **Get all product in Cart (POST) (/api-get-all-product-in-cart)**
   1. Request
3. Request Body

{

“userID”: “idabc123”

}

* 1. Response

1. Success

{

“code” : CODE.SUCCESS,

“message” : MESSAGE.SUCCESS,

“listProductInCart”: [

{

“id” : “xyz456”,

“name”: “PUBG”,

“price” : 150000,

“des” : “Game of the year”,

“category”: “steam”,

“image”: “456”,

“quantity”: 2

},

{

“id” : “xyz457”,

“name”: “CSGO”,

“price” : 160000,

“des” : “Game of the year”,

“category”: “origin”,

“image”: “457”,

“quantity”: 1

}

]

}

1. UserID not found

{

“code” : CODE.DATA\_NOTFOUND,

“message” : MESSAGE.DATA\_NOTFOUND\_USERID

“content” : null

}

1. **Update product in cart (POST) (/api-update-product-in-cart)**
   1. Request
2. Request Body

{

“userID”: “idabc123”,

“productID” : “xyz456”,

“type” : “add”

}

-------------------------------

{

“userID”: “idabc123”,

“productID” : “xyz456”,

“type” : “remove”

}

* 1. Response

1. Success

{

“code” : CODE.SUCCESS,

“message” : MESSAGE.SUCCESS,

“content” : null

}

1. UserID not found

{

“code” : CODE.DATA\_NOTFOUND,

“message” : MESSAGE.DATA\_NOTFOUND\_USERID

“content” : null

}

1. ProductID not found

{

“code” : CODE.DATA\_NOTFOUND,

“message” : MESSAGE.DATA\_NOTFOUND\_PRODUCTID

“content” : null

}

1. Product is not in cart to be removed (Không có sp trong cart nhưng gửi lệnh remove)

{

“code” : CODE.DATA\_NOTFOUND,

“message” : MESSAGE.DATA\_NOTFOUND\_PPRODUCTINCART

“content” : null

}

1. **Payment:**
2. **Checkout\_with\_new\_info (POST) (/api-checkout-with-new-info)**
   1. Request:
3. Request body:

{

“type”:”0,

“userID”: “idabc123”,

“name”: “nguoinhanhang”,

“address”:”Diachinhanhang”,

“email” : “[emailnguoinhan@gmail.com](mailto:emailnguoinhan@gmail.com)”,

“note”: “giaohanggiohanhchinh” // allow null

}

* 1. Response:

1. Success

{

“code” : CODE.SUCCESS,

“message” : MESSAGE.SUCCESS,

“content” : null

}

1. UserID not found

{

“code” : CODE.DATA\_NOTFOUND,

“message” : MESSAGE.DATA\_NOTFOUND\_USERID

“content” : null

}

1. Cart is empty

{

“code” : CODE.DATA\_NOTFOUND,

“message” : MESSAGE.DATA\_NOTFOUND\_PPRODUCTINCART

“content” : null

}

1. **Checkout\_with\_default\_info (POST) (/api-checkout-with-default-info)**
   1. Request
2. Request body:

{

“type”:”1,

“userID”: “idabc123”

}

1. Success

{

“code” : CODE.SUCCESS,

“message” : MESSAGE.SUCCESS,

“content” : null

}

1. UserID not found

{

“code” : CODE.DATA\_NOTFOUND,

“message” : MESSAGE.DATA\_NOTFOUND\_USERID

“content” : null

}

// Thanh toán có 2 option 1 sử dụng thông tin mới 2 sử dụng địa chỉ mặc định

// Nếu thanh toán thành công server thực hiện thêm các thông tin của người nhận (

“name”: “nguoinhanhang”,

“address”:”Diachinhanhang”,

“email” : “[emailnguoinhan@gmail.com](mailto:emailnguoinhan@gmail.com)”,

“idsteam”: “dsteamnguoinhan”,

“note”: “giaohanggiohanhchinh” // allow null

) , thời gian đặt hàng vào đối tượng Cart của user id đó và sửa type của Cart thành Order và cập nhật vào DB

Đồng thời cũng tạo ra Cart mới với type là cart gắn với User đó vì Cart cũ đã update thành Order