

# Vani Development Specification

Kyu Hou – houkyu@vanilacorp.com – Version 1.1.4, 2023-06-23

---

## Table of Contents

Introduction

Version History

Ground Rule for API

- API Protocol

- Security

- API Versioning

- API Performance

API Errors Handling

- Http Status Code

- Error Response

Overview (Membership - Offline Model)

- Membership Integration

- Barcode Usage

Overview (Membership - Online Model)

- vani App Connection for non vani user

- Membership Integration

- Online Transaction

Overview (Point Swap)

- Point Swap

REST API for Membership (vani → Membership)

- API Guideline for Membership System

- Checking Membership Customer

- Connection between vani and a Membership

- Membership Registration

- Notifying vani user's information changes

- Searching Membership Information

- Point history search

- vani disconnection

- Issuing Coupon (Optional)

- Checking Coupon Status (Optional)

REST API for Membership (Membership → vani)

- Server Information

- Notifying Membership Deregistration

- Searching vani Barcode Number (Optional)

- Delivering Membership Point Transaction Data (realtime)

- Delivering Membership Point Transaction Data (Bulk Insert)

- Delivering Membership Point Cancellation data

- Delivering Membership Point Cancellation data(Bulk Insert)

REST API for Point Swap (vani → Membership )

- Checking Remain Amount of Points

- Membership Points Refill

- Membership Point Usage

Cancellation of Membership Point refill or Usage  
Transaction Information of Membership Point  
Web page link (Only for Online Merchant)  
REST API for Filling Vani Coin(point) (Merchant → vani)  
Server Information  
Technical considerations  
filling vani coin(point)  
Cancellation of Filling Transaction  
Transaction Information of Points Filling

---

Copyright © Vanila Studio Co., Ltd. All Rights Reserved.

## Introduction

The default language for this document is English. Vietnamese and Korean is also supported if necessary.

[KO] in front of a sentence indicates that the following sentence is Korean translation of previous sentence. Likewise [VN] indicates Vietnamese

[VN]: Ngôn ngữ mặc định cho tài liệu này là tiếng Anh. Từ khóa [VN] đứng trước nội dung được chuyển ngữ sang tiếng Việt khi cần thiết.

[KO]이 문서의 기본언어는 영어입니다. 또한, 필요한 경우 베트남어와 한글 번역을 제공합니다. 문장이 [KO]로 시작하는 경우 이전 문장의 한글 번역 내용입니다. 마찬가지로 문장이 [VN]으로 시작하는 경우는 베트남 번역 내용입니다.

This document contains a detailed guide on the APIs that the vani System and 3rd Party Membership System should develop for vani Service.

[VN]Văn bản này bao gồm hướng dẫn cụ thể cho vani System và 3rd Party Membership System (đối tác Membership) của vani nhằm mục đích phát triển APIs cho vani Service.

[KO]이 문서는 vani Service를 위해 vani System과 vani와 제휴한 외부 Membership System이 개발해야되는 API에 대한 세부 Guide를 포함합니다.

This document is subject to continuous change as requested by vani Service evolution or a 3rd Party Membership. Always refer to the latest version of the document.

[VN]Văn bản này được liên tục cập nhật theo yêu cầu về sự phát triển về dịch vụ của vani hoặc yêu cầu của các bên Thành viên thứ 3. Hãy liên tục cập nhật phiên bản mới nhất của văn bản này.

[KO]이 문서는 vani Service 진화 또는 외부 Membership 사의 요청에 따라 지속적으로 변경될 수 있습니다. 항상 최신 Version의 문서를 참고하시기 바랍니다.

## Version History

It records the version and changes of documents that vani officially distributes.

Version	Changes	Release Date(YYYY-MM-DD)
1.0.0	Initial Release. English & Korean support	2021-09-23
1.0.1	A request parameter changed for Point history search	2021-10-07
1.0.2	isVanilaBarcodeUsed parameter added at /points-approval and /points-approval-bulk API	2021-11-10
1.0.3	addressDetailInfo deleted in the response body of /check-user API	2021-12-17
1.0.4	DUP_EMAIL error code defined in the /registration API	2021-12-17
1.0.5	Vanila System's domain name changed (vanila.me → vani.la)	2021-12-30
1.0.6	Vanila System's host name changed ( membership-test → membership-api.test , membership → membership-api )	2022-2-15
1.0.7	reason added in the request body of /disconnection API. totalPointAmount added in the response body of /points-refill and /points-usage API. branch and totalPointAmount changed to mandatory field in /points-approval and /points-approval-bulk API.	2022-02-25

Version	Changes	Release Date(YYYY-MM-DD)
1.0.8	/points-cancellatioin-bulk API added	2022-03-07
1.0.9	Vietnamese Language added TIPs for /points-history-search added	2022-03-08
1.0.10	Overview of online model added Name changed from Vanila to vani ( API spec is not affected)	2022-06-08
1.0.11	trafficType added at /points-approval and /points-approval-bulk API due to distinguishing between online traffic and offline traffic	2022-07-06
1.0.12	- Deleted the positive integer info at pointAmount in /points-approval, /points-approval-bulk, meaning that pointAmount can be a floating point - Added the positive integer info at refillPoints exchangedVanilaPoints in /points-refill, meaning that the exchanged points should be integer - Added the positive integer info at usagePoints in /points-usage, meaning that the exchanged points should be integer	2022-07-18
1.0.13	- Added /coupon-issuance and /check-status-coupon API for issuing merchant's coupon and checking the coupon status	2022-09-01
1.0.14	- Added /points-filling, /points-filling-cancellation and /points-filling-info/{requestId} API for points filling releated transaction. - amountOfVaniCoin added in /barcode-search API	2022-09-13
1.0.15	- Changed iss value to membershipId in /points-filling, /points-filling-cancellation API - eventType property added in request payload in /points-filling API	2022-09-30
1.0.16	- Changed paymentMethod, productTitle, productAmount, productCurrency property to mandatory data in /points-approval and /points-approval-bulk API	2022-12-05
1.0.17	- Added the billAmount in /points-approval, points-approval-bulk API	2023-02-16
1.0.18	- Added the comments on customerId and other fields	2023-03-23
1.1.0	- Launched new api for point history and deleted previous api. ( deleted /points-history-search API and added /points-history-details API )	2023-04-04
1.1.1	- Added vaniCouponNumber in /points-approval and /points-approval-bulk API for receiving Vani's coupon number	2023-04-28

Version	Changes	Release Date(YYYY-MM-DD)
1.1.2	- Added countryCode , idType ,idNumber in /registration and /user-info-renewal API for delivering nationality and ID type and Number of a Vani user to a merchant	2023-06-01
1.1.3	- Added max length for some parameters of API request	2023-06-02
1.1.4	- Added /secure-registration API with encryption mechanism (AES-GCM) - Deleted countryCode , idType ,idNumber in /registration	2023-06-23

## Ground Rule for API

The API provided by vani and the API provided by Membership must follow the rules below.

[VN]API được cung cấp bởi vani và API được cung cấp bởi đối tác Membership phải tuân theo những quy định sau.

[KO]vani 에서 제공하는 API와 Membership사에서 제공하는 API는 아래 Rule 따라야 합니다.

### API Protocol

- All APIs must be provided based on JSON over HTTP.
  - Content-Type is `application/json` and charset is `utf-8`.
- All API must support HTTPS.
  - Membership System must support TLS 1.2 or higher protocol for secure data transmission.

### Security

- API requests should be authenticated via Basic Authentication. Therefore the request should include HTTP Authorization Header with Access Token issued by API owner which would be vani or 3rd Party Membership.

```
Authorization: Basic {Access Token}
```

JAVASCRIPT

- For security reasons, 3rd Party Membership should provide a domain name( ex. mydomain.com ) to the vani system. Vanils System manages the information.

### API Versioning

The version of the API exists as vX.X after the Base URL. The current API version is v1.0.

- The full URL configuration is {Base URL}/{api-version}/{endpoint} , so if the base URL is `membership.yourdomain.com` , the api-version is `v1.0` , and the endpoint is `disconnection` , the full URL configuration is as follows.

Sample URL

<https://membership.yourdomain.com/v1.0/disconnection> (https://membership.yourdomain.com/v1.0/disconnection)



The **base URL** provided by vani System is defined in this documentation. 3rd Party Membership should provide **base URL** information to vani.

### API Performance

Every API response time should be no longer than 10 seconds.

- In case when the API response time exceeds 10 seconds for a certain period of time, Vani System would pause requesting the API for a couple of minutes and then request again.
- If the delay continues, vani would notify the merchant and ask for fixing it.

# API Errors Handling

Success or failure of all API requests should be known by HTTP Status Code .

[VN]Thành công hay thất bại của tất cả các API request phải được xác nhận bằng HTTP Status code .

[KO]모든 API request에 대한 성공과 실패 여부는 HTTP Status Code 로 확인할 수 있어야 합니다.

## Http Status Code

HTTP Status Codes and their meanings are as follows.

HTTP Status Code	Meaning
200,201	Success
400	BAD-REQUEST - the request is invalid
401	UNAUTHORIZED - API Token is not valid
403	FORBIDDEN - API Token is valid but no authorization
404	NOT FOUND - No resource
500,502,503,504	SERVER ERROR - something wrong with Server

## Error Response

When an error occurs (HTTP Status Code is not 2XX), the following Error Object should be delivered in Json format.

[VN]Khi có lỗi (HTTP Status Code không phải là 2XX), Error Object phải được truyền tải dưới dạng Json format.

[KO]API 요청에 대해 성공 (HTTP Status Code 가 2XX)이 아닌 경우는 아래 Json Format으로 Error 정보를 전달해야 합니다.

### Error Object

Property	Type	Always Exists	Description
errorCode	string	Yes	Server defined Error Code.
errorMessage	string	Yes	A detailed explanation of the error code. errorMessage <b>must be written in English</b> .

### Sample Error Response

```
{
  "errorCode": "INSUFFICIENT_PARAMTER",
  "errorMessage": "mobilePhoneNumber is mandatory"
}
```

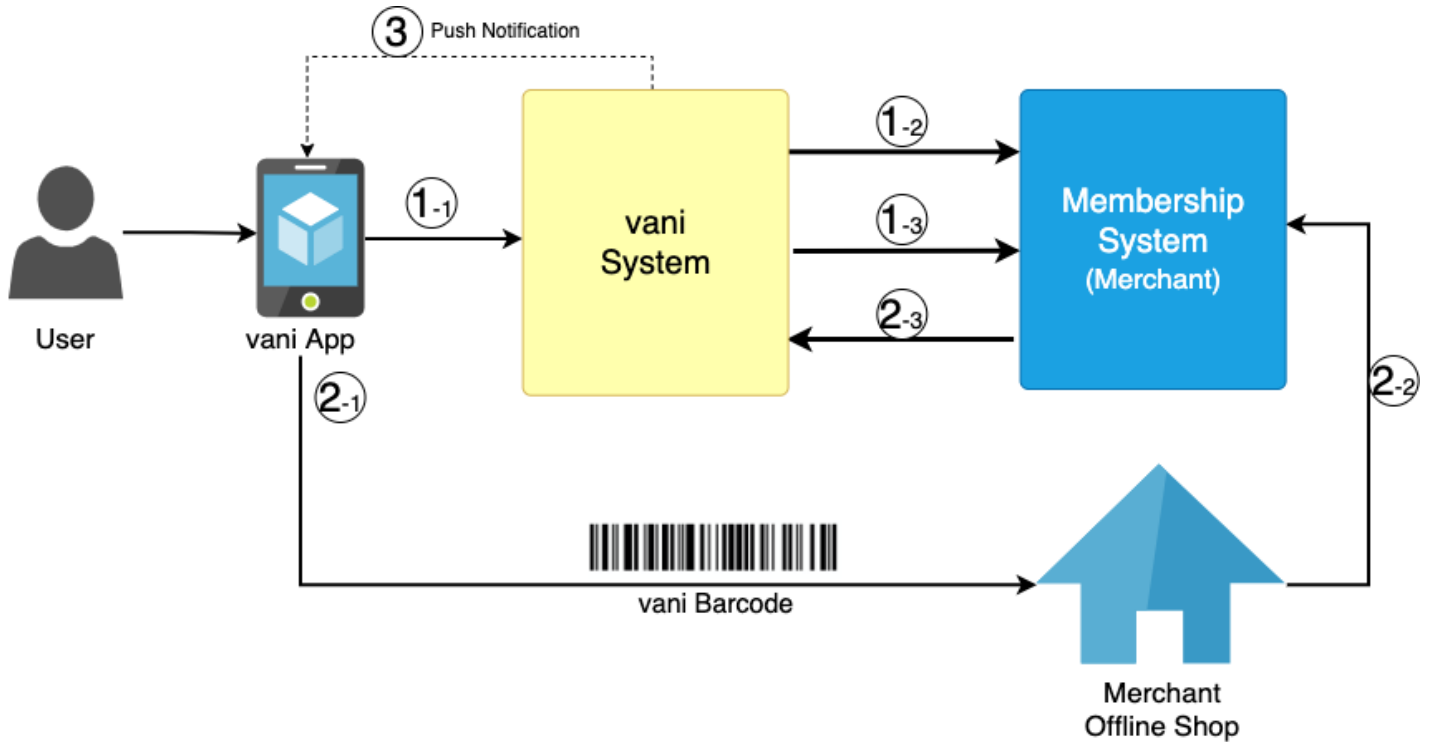
JSON

## Overview (Membership - Offline Model)

This section describes the Main Scenario that must be linked with the vani System in the process of membership registration and use.

[VN]Phần này miêu tả những Main Scenario phải kết nối với vani System trong quá trình đăng ký và sử dụng membership.

[KO]membership 등록 및 사용 과정에서 vani System과 연동해야 되는 Main Scenario를 설명합니다.



## Membership Integration

### Step 1-1

Customers select a Membership (ex. L Point) that requires registration through the vani App.

[VN]Khách hàng chọn chương trình thành viên (vd. L Point) cần đăng ký qua ứng dụng vani App

[KO]고객은 vani App을 통해 등록이 필요한 Membership(ex. L point)을 선택합니다.

### Step 1-2

The vani System verifies whether the membership system has customer information based on the customer information (mobile phone number, date of birth) held by vani. If there is(are) matched customer(s), the Membership System responds with additional information such as name and membership card number.

[VN]vani System xác minh Membership System đó có thông tin khách hàng hay không dựa trên thông tin khách hàng (số điện thoại di động, ngày sinh) do vani nắm giữ. Nếu thông tin khách hàng trùng khớp, Membership System sẽ phản hồi với thông tin bổ sung như tên và số thẻ thành viên

[KO]vani System은 vani가 보유하고 있는 고객정보 (휴대폰 번호, 생년월일) 기준으로 Membership System의 고객정보 보유 여부를 확인 합니다. Membership System은 전달받은 정보와 일치하는 고객정보가 있을 경우 이름, 멤버십 번호 등 추가정보를 포함하여 전달합니다.

### Step 1-3

vani System utilizes the information received in Step 1-2 to transmit the barcode information of vani if there is a customer with the same name, mobile phone number, and date of birth to link customer information with each other. If there is no matching customer, the vani system requests registration for membership using the customer information you have.

[VN]Trong trường hợp thông tin khách hàng trùng khớp, ví dụ như trùng tên, số điện thoại, ngày tháng năm sinh, vani System sẽ sử dụng thông tin nhận được ở Bước 1-2 để truyền thông tin mã vạch của vani để liên kết thông tin khách hàng giữa 2 hệ thống với nhau. Nếu thông



tin khách hàng không trùng khớp, hệ thống vani sẽ yêu cầu đăng ký thành viên bằng thông tin của khách hàng trong hệ thống của vani

[KO]vani System은 Step 1-2에서 전달받은 정보를 활용하여 이름, 휴대폰 번호, 생년월일 모두 일치하는 고객이 있으면 vani의 barcode 정보를 전달하여 상호 고객정보 연동을 수행합니다. 만약 일치하는 고객이 존재하지 않으면 vani system은 보유한 고객정보를 이용하여 membership에 registration을 요청 합니다.



Merchant's Membership System must store the vani Barcode delivered by the vani System.

[VN]Merchant của Membership System phải lưu dữ liệu về vani Barcode được chuyển đến từ vani System.

[KO]Merchant의 Membership System은 vani System에서 전달한 vani Barcode를 저장해야 합니다.

## Barcode Usage

### Step 2-1

Customers use the vani Barcode created in the vani App to request earn or redeem points of membership in the Offline Shop.

[VN]Khách hàng dùng vani Barcode được tạo từ ứng dụng vani để tích điểm hoặc sử dụng điểm thành viên tại cửa hàng của đối tác (Offline Shop).

[KO]고객은 vani App에 생성된 vani Barcode를 이용하여 Offline Shop에 Membership 사용 또는 적립 요청을 합니다.

### Step 2-2

The Offline Shop passes the Vanlia Barcode received through POS to the Membership System.

[VN]Offline Shop sử dụng POS để truyền tải vani Barcode đến Membership System.

[KO]Offline Shop은 POS를 통해 전달받은 Vanlia Barcode를 Membership System에 전달합니다.

### Step 2-3

Membership System finds customers who correspond to Vanlia barcode, processes points earning or redemption, and delivers the results to the vani System in real time.

[VN]Membership Systems tìm khách hàng tương ứng với vani Barcode, xử lý điểm cần tích hoặc sử dụng, và gửi kết quả đến vani System trong thời gian thực.

[KO]Membership System은 vani Barcode에 해당하는 고객을 찾아서 포인트 적립 또는 사용을 처리하고 그 결과를 실시간으로 vani System으로 전달합니다.

### Step 3

vani System sends a Push Notification to vani App based on the received data.

[VN]vani System gửi Push Notification đến ứng dụng vani dựa theo thông tin nhận được.

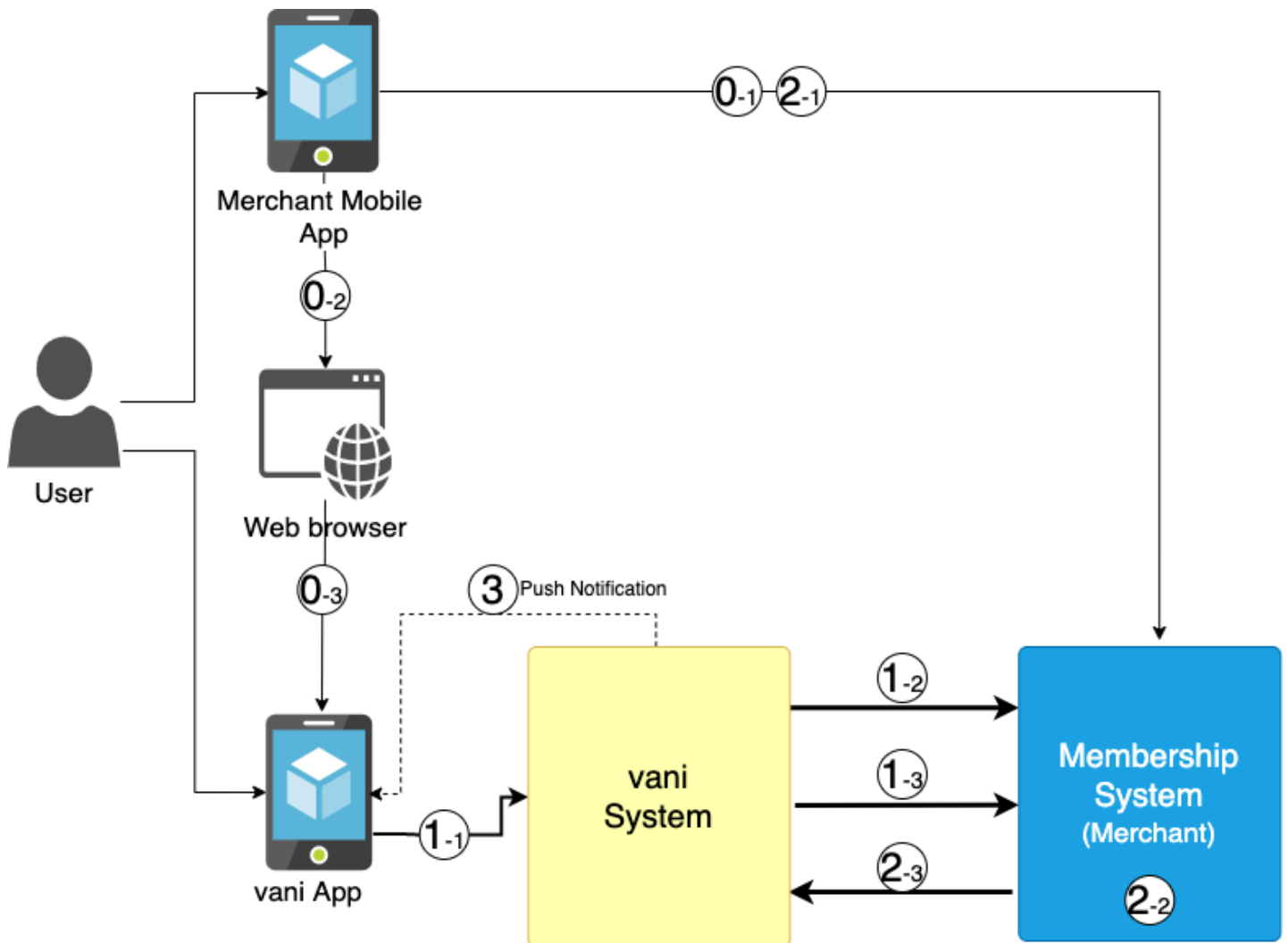
[KO]vani System은 전달받은 data를 바탕으로 vani App에 Push Notification을 보냅니다.

## Overview (Membership - Online Model)

This section describes the Main Scenario that must be linked with the vani System in the process of membership registration and use as well as interaction with vani application.

[VN]Phần này miêu tả những Main Scenario phải liên kết với Vanila System và quá trình liên kết với ứng dụng vani khi người dùng đăng ký và sử dụng thẻ thành viên.

[KO]membership 등록 및 사용 과정에서 vani System과 연동해야 되는 Main Scenario와 vani App과의 연결과정을 설명합니다.



### vani App Connection for non vani user

#### Step 0-1

When a Merchant App customer enters a page that promotes vani, the Merchant App checks whether this customer is a vani customer from the Merchant Server.

[VN]Khi khách hàng của Merchant App đi vào trang tiếp thị cho vani, Merchant App sẽ bắt đầu kiểm tra xem khách hàng này có phải là người dùng của vani hay không từ Merchant Server.

[KO]Merchant App 고객이 vani 홍보를 하는 Page에 진입할 경우, Merchant App은 이 고객이 vani 고객 인지를 Merchant Server로 부터 확인 합니다.

#### Step 0-2

If the customer of Merchant is not a customer of vani, the Merchant App calls the URL provided by vani in the browser.

[VN]Trong trường hợp khách hàng của Merchant không phải là người dùng của vani, Merchant App sẽ gọi URL được vani cung cấp tại

browser.

[KO]Merchant의 고객이 vani 고객이 아닐 경우, Merchant App은 vani에서 제공한 URL을 browser에서 호출 합니다.



For proper operation of the page provided by vani, it is recommended to call it from the browser, not the webview.

[VN]Để các thao tác của trang được vani cung cấp diễn ra thuận lợi, nên gọi tại browser chứ không phải webview.

[KO]vani에서 제공하는 page의 적절한 동작을 위해 webview가 아닌 browser에서 호출 하는것을 권고합니다.

## Membership Integration

### Step 1-1

This process is same as Step1-1 of Membership Intergration Offline Model

[VN]Giống với quy trình của Step1-1 của Membership Integration Offline Model

[KO]Membership Intergration Offline의 Step1-1 과 과정이 동일합니다.

### Step 1-2

This process is same as Step1-2 of Membership Intergration Offline Model

[VN]Giống với quy trình của Step1-2 của Membership Integration Offline Model

[KO]Membership Intergration Offline의 Step1-2 과 과정이 동일합니다.

### Step 1-3

This process is same as Step1-3 of Membership Intergration Offline Model

[VN]Giống với quy trình của Step1-3 của Membership Integration Offline Model

[KO]Membership Intergration Offline의 Step1-3 과 과정이 동일합니다.

## Online Transaction

### Step 2-1

When a purchase is made in the Merchant App, the purchase details are transmitted to the Merchant Server.

[VN] Lịch sử giao dịch sẽ được gửi về Server của Merchant khi phát sinh giao dịch tại Merchant App.

[KO]Merchant App에서 구매행위가 이루어지면 구매내역이 Merchant의 Server로 전달됩니다.

### Step 2-2

Merhant Server determines whether the customer is a vani user.

[VN]Merchant Server sẽ phán đoán xem liệu khách hàng giao dịch có phải là người dùng của vani hay không

[KO]Merhant Server는 구매고객이 vani 사용자 인지 판단합니다.

### Step 2-3

Merchant's Membership System handles the payment and point transaction internnlay and if the customer of the transaction is a vani customer, the point processing details including transaction details should be transmitted to the vani System in real time.

[VN]Sau khi xử lý tích hoặc sử dụng điểm cho khách hàng, nếu khách hàng này là người dùng của vani, Membership System của Merchant sẽ gửi lịch sử xử lý điểm có bao gồm cả lịch sử giao dịch về vani System theo thời gian thực.

[KO]Merchant의 Membership System은 포인트 적립 또는 사용을 처리하고 해당 거래의 고객이 vani 고객인 경우 거래내역을 포함한 포인트 처리 내역을 실시간으로 vani System으로 전달합니다.

### Step 3

vani System sends a Push Notification to vani App based on the received data.

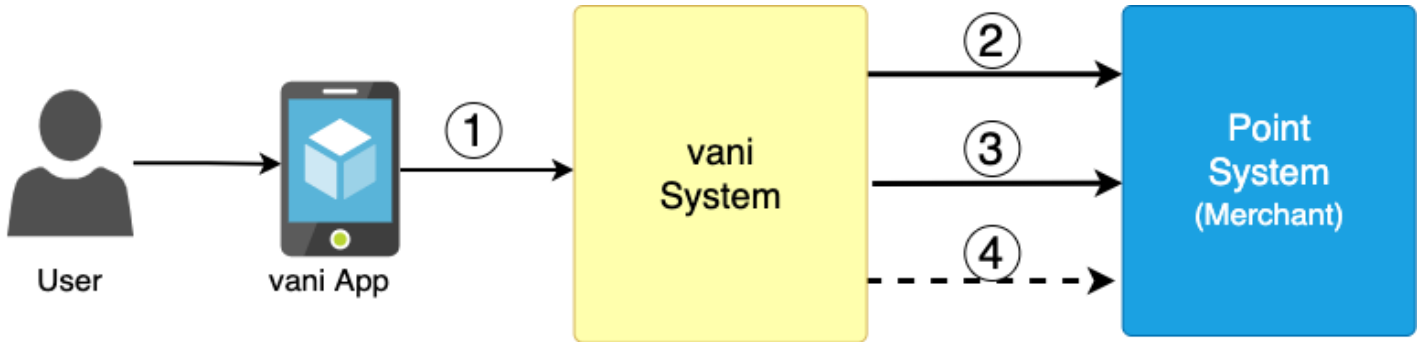
[VN]vani System gửi Push Notification đến ứng dụng vani dựa theo thông tin nhận được.

[KO]vani System은 전달받은 data를 바탕으로 vani App에 Push Notification을 보냅니다.

## Overview (Point Swap)

vani Points are interchangeable with Merchant Points.

[KO]vani Point는 Merchant의 Point와 상호 교환할 수 있습니다.



## Point Swap

### Step 1

A customer selects the membership for point swap through the vani App.

[VN]Điểm vani có thể đổi với điểm của Merchants.

[KO]고객은 vani App을 통해 Point Swap을 원하는 membership을 선택합니다.

### Step 2

vani System searches points based on customer information (customer id) of the membership received when linking with the membership.

[VN]Hệ thống vani tra cứu điểm dựa trên thông tin khách hàng (id khách hàng) mà bên membership system cung cấp khi liên kết với hệ thống thành viên giữa 2 bên.

[KO]vani System은 membership system과 연동 시 전달받은 membership의 고객정보(customer id)를 기반으로 point를 조회 합니다.

### Step 3

Customers request to convert vani Points to Membership Points. When the conversion request is completed, vani Points will be deducted.

[VN]Khách hàng yêu cầu chuyển đổi điểm vani thành điểm Membership Points. Khi yêu cầu chuyển đổi được hoàn thành, tổng số điểm vani sẽ bị trừ một khoản tương ứng.

[KO]고객은 vani Point를 Membership Point로 전환 요청 합니다. 전환 요청이 완료 되면 vani Point는 차감됩니다.

### Step 4

If vani System does not receive a response from Merchant's Point System in the process of Step 3 , it checks whether the request was processed or not.

[VN]Nếu hệ thống vani không nhận được phản hồi từ Hệ thống điểm của Merchant trong bước thứ 3, vani sẽ kiểm tra xem yêu cầu đã được xử lý chưa.

[KO]만약 Step 3 과정에서 vani System이 Merchant의 Point System으로 부터 응답을 받지 못했을 경우, 정상 처리 여부를 조회하여 결과에 따라 적절한 처리를 수행합니다.



vani System waits up to 10 seconds for a response to a Point Swap request.

[VN]vani System đợi tối đa 10 giây cho phản hồi yêu cầu hoán đổi điểm.

[KO]vani System은 Point Swap 요청에 대해 최대 10초 동안 응답을 기다립니다.

## REST API for Membership (vani → Membership)

This section describes the API that the **Membership System should provide for membership integration** with the vani System.

[VN]Phần này giải thích về API mà Membership System cần cung cấp để tích hợp hệ thống Membership với vani System.

[KO]Membership System이 vani System과의 멤버십 연동을 위해 제공해야 되는 API에 대해 설명합니다.

### API Guideline for Membership System

- Merchant's Membership System should comply with Ground Rule, API Errors Handling defined in this documentation.

[VN]Membership System của Merchant cần tuân thủ Ground Rule, API Errors Handling theo quy định được giải thích trong văn bản này.

[KO]Merchant의 Membership System은 본 규격서가 정의하는 Ground Rule, API Errors Handling 부분을 준수해야 합니다.

- Membership System must deliver the base URL and api-version information of the provided API.

[VN]Membership System phải gửi Base URL và thông tin api-version của API được cung cấp.

[KO]Membership System은 제공하는 API 의 Base Url과 api-version정보를 전달해야 한다.

- vani System calls Membership System's API by composing {Base URL}/{api-version}/{endpoint} as defined in API Versioning.

[VN]vani System gọi Membership System bằng cách soạn URL API ({Base URL} / v1.0 / {endpoint}) như được định nghĩa trong API Versioning.

[KO]vani System은 API Versioning에 정의된 방식으로 API URL을 구성({Base URL}/{api-version}/{endpoint}) 하여 Membership System을 호출한다.

- If the base URL is `mymembership.com`, the api-version is `v1.0`, and the endpoint is `disconnection`, vani System constructs the following URL and requests it to the Membership System.

[VN]Nếu base URL là `mymembership.com`, api-version là `v1.0` và endpoint là `disconnection`, vani System soạn nội dung URL như dưới đây và gửi yêu cầu tới Membership System.

[KO]만약 Base URL이 `mymembership.com`, api-version이 `v1.0`, endpoint가 `disconnection` 이면 vani System은 아래 URL을 구성하여 Membership System에 요청한다.

<https://mymembership.com/v1.0/disconnection> (https://mymembership.com/v1.0/disconnection)

### Checking Membership Customer

Based on vani user's the mobile phone number and birthday, vani System checks whether the user is also a customer of the 3rd Party Membership.

[VN]Dựa vào thông tin về số điện thoại và ngày sinh của khách hàng của vani, vani Systems xác nhận người dùng đó có phải cũng là khách hàng của đối tác chương trình Thành viên hay không.

[KO]vani에서 보유한 휴대폰 번호와 고객생일을 기준으로 고객이 연결하고자 하는 Membership 가입여부를 확인합니다.

#### Endpoint

POST /check-user

#### JSON Structure

##### Request Body

Property	Type(Max Len)	Necessity	Description
membershipId	string(10)	Required	Membership id generated by vani

Property	Type	Type (Max Len)	Necessity	Description
matchingType	string	10	Required	Mobile phone number based matching (Fixed Value)
matchingValue	string	(20)	Required	Mobile phone number if matchingType is PHONE. Digit only(ex. 0709022117)
dateOfBirth	string	(10)	Required	Customer's birthday. format: YYYY-MM-DD ( <u>ISO-8601</u> (https://www.w3.org/TR/NOTE-datetime) )

- Case 1) If there is(are) Matching customer(s)
  - [KO]양 사 간의 고객정보가 일치하는 고객이 있을 경우
  - [VN]Nếu có thông tin khách hàng trùng khớp giữa hai hệ thống

Response Body

Property	Type	Always Exists	Description
customerInfoList	array	Yes	Matched customer information
customerId	string	Yes	A Unique identifier of a customer (generated by Membership System)
customerName	string	Yes	Customer Name (Full Name)
gender	string	Yes	Gender M : male, F : female, U :unknown
brandBarcode	string	Yes	Membership card number

Sample

Request

```
{
  "membershipId": "BRAND1",
  "matchingType": "PHONE",
  "matchingValue": "0321231234",
  "dateOfBirth": "1990-12-30"
}
```

JSON

Response

```
{
  "customerInfoList": [
    {
      "customerId": "uuser-adfadc-dfea",
      "customerName": "HONG GIL DONG",
      "gender": "M",
      "brandBarcode": "1234123412341234"
    }
  ]
}
```

JSON



**customerId** **MUST NOT** be one of the user's information, such as mobile phone number or vanillaBarcode.

[VN] **customerId** không thể là thông tin cá nhân (Số điện thoại hoặc Mã vạch Vani)

[KO] `customerId`는 고객정보 (휴대폰 번호나 Vanilla 바코드)가 되어서는 안됩니다.

- Case 2) If there is(are) no matching customer(s)

[VN]Nếu không có thông tin khách hàng trùng khớp

[KO]양 사 간의 일치하는 고객이 없을 경우

### Response Body

None



If no matching customer is found, the HTTP Status Code should be passed as **404**.

[VN]Nếu không tìm thấy khách hàng phù hợp thì HTTP status Code sẽ chuyển sang **404**.

## Connection between vani and a Membership

If the customer information owned by vani System and Membership System is identical, vani System sends vani's barcode number to the Membership System

[VN]Nếu thông tin của khách hàng của vani System và Membership System trùng khớp, vani System gửi thông tin mã vạch vani đến Membership System.

[KO]vani가 보유한 고객정보가 membership system에서 보유한 고객정보와 일치하는 경우 vani가 생성한 바코드 번호를 전달합니다.



Membership System should save the vani's barcode number information.

[VN]Membership System phải lưu thông tin số mã vạch của vani.

[KO]membership system은 vani가 전달하는 바코드 번호를 저장해야 합니다.



The Membership system must ensure that this API is idempotent. In other words, vani System can call this API multiple times when it does not receive a response due to network disconnection. In this case, Membership System should always deliver the same result

[VN]Membership System phải đảm bảo API này idempotent. Nói cách khác, vani System có thể gọi API này nhiều lần khi không nhận được phản hồi bình thường do bị ngắt kết nối mạng, v.v. Trong trường hợp này, Membership System phải luôn truyền đi cùng một kết quả.

[KO]Membership system은 이 API의 멍등성을 보장해야 합니다. 즉, vani SYstem은 네트워크 연결끊김 등으로 정상적인 응답을 못 받았을 때 이 API를 여러번 호출 할 수 있습니다. 이런경우, Membership System은 항상 동일한 결과를 전달해야 합니다.

### Endpoint

POST /connection

### JSON Structure

#### Request Body

Property	Type(Max Len)	Necessity	Description
requestId	string(200)	Yes	Unique request id generated by vani
membershipId	string(10)	Yes	Membership ID generated by vani

customerId	string	Yes	A Unique identifier of a customer generated by Membership System
vanilaBarcode	string(16)	Yes	vani's barcode number. 16 dit

### Response Body

Property	Type	Always Exists	Description
customerId	string	yes	A Unique identifier of a customer generated by Membership System
membershipLevel	string	Yes	Membership level (ex. VIP , GOLD ). NONE if level doesn't exist.
membershipPoint	number	Yes	Membership Point of the customer



**customerId** **MUST NOT** be one of the user's information, such as mobile phone number or vanilaBarcode.

[VN] **customerId** không thể là thông tin cá nhân (Số điện thoại hoặc Mã vạch Vani)

[KO] `customerId`는 고객정보 (휴대폰 번호나 Vanila 바코드)가 되어서는 안됩니다.

## Membership Registration

If there is no matching customer information between vani and Merchant, membership registration is requested using vani's customer information.

A merchant can choose basic API( /registration ) or encrypted API( /secure-registration ) based on which user information the merchant is necessary.

[VN]Nếu không có thông tin khách hàng trùng khớp giữa vani và Merchant, gửi yêu cầu đăng ký thành viên bằng thông tin khách hàng của vani. Một merchant có thể chọn API cơ bản (/registration) hoặc API mã hóa (/secure-registration) dựa trên những thông tin người dùng mà merchant cần.

[KO]vani에서 Merchant간 일치하는 고객정보가 없는 경우, vani의 고객정보를 이용하여 Membership 등록을 요청 합니다. Merchant가 필요한 고객정보의 성격에 따라 기본 API( /registration ) or 또는 암호화 처리된 API( /secure-registration )를 선택할 수 있습니다.



The Membership system must ensure that this API is idempotent. In other words, vani System can call this API multiple times when it does not receive a response due to network disconnection. In this case, Membership System should always deliver the same result

[VN]Membership System phải đảm bảo API này idempotent. Nói cách khác, vani System có thể gọi API này nhiều lần khi không nhận được phản hồi bình thường do bị ngắt kết nối mạng, v.v. Trong trường hợp này, Membership System phải luôn truyền đi cùng một kết quả.

[KO]Membership system은 이 API의 멍등성을 보장해야 합니다. 즉, vani SYstem은 네트워크 연결끊김 등으로 정상적인 응답을 못 받았을 때 이 API를 여러번 호출 할 수 있습니다. 이런경우, Membership System은 항상 동일한 결과를 전달해야 합니다.

## Endpoint

POST /registration

This API is for a merchant which does not require ID Card Number /Nationality during registration.

[VN]API này dành cho merchant không yêu cầu số CMND/ Quốc tịch trong quá trình đăng ký.

[KO]이 API는 ID Card 번호(여권번호 포함) 나 국적정보가 필요하지 않을 경우에 적용할 수 있습니다.



### Request Body

Property	Type(Max Len)	Necessity	Description
requestId	string(200)	Yes	Unique request id generated by vani
membershipId	string(10)	Yes	Membership id generated by vani
mobilePhoneNo	string(20)	Yes	Mobile phone number
dateOfBirth	string(10)	Yes	Customer's birthday. format: YYYY-MM-DD ( <u>ISO-8601</u> ( <a href="https://www.w3.org/TR/NOTE-datetime">https://www.w3.org/TR/NOTE-datetime</a> ) )
gender	string(20)	Yes	Gender. M : Male, F : Female
customerName	string(512)	Yes	Customer name (Full Name)
emailAddress	string(256)	No	Email Address
vanilaBarcode	string(16)	Yes	vani's barcode number. 16 dit
addressDetailInfo	object	No	Detail address information
state	string(500)	No	State (tỉnh / thành phố)
district	string(500)	No	District (Quận / Huyện)
detail1	string(500)	No	Detail address part1 면/동 (Phường/Xã)
detail2	string(500)	No	Detail address part2

### Response Body

Response Body

### Error Code

Error Code

### Endpoint

POST /secure-registration

This API is for a merchant which does requires ID Card Number /Nationality during registration.

[VN]API này dành cho merchant yêu cầu số CMND/ Quốc tịch trong quá trình đăng ký.

[KO]이 API는 ID Card 번호(여권번호포함)와 국적정보가 필요한 경우 적용할 수 있습니다.

- The original request body (JSON) should be encrypted in the security mechanism.  
[VN]Nội dung yêu cầu ban đầu (JSON) phải được mã hóa theo cơ chế bảo mật.  
[KO]요청정보(JSON)는 암호화 적용해야 합니다.
  - Security Mechanism : AES-GCM (Advanced Encryption Standard-Galois/Counter Mode)
    - Associate Data and Authentication Tag are not used because it's optional.  
[VN]Thẻ Associate Data và Authentication không được sử dụng vì đây là tùy chọn.  
[KO] Associate Data 와 Authentication 은 Option 이므로 사용하지 않습니다.

- Vani will share the encryption key to the merchant.  
[VN]Vani sẽ chia sẻ khóa mã hóa cho merchant.  
[KO]Vani 는 암호화를 위한 Key를 생성해서 제휴사에 전달합니다.
- The encryption key for testing and production environments are different.  
[VN]Khóa mã hóa cho môi trường test và môi trường production là khác nhau.  
[KO]암호화 키는 개발/운영 환경에 서로다른 키가 각각 부여됩니다.

Request Data

Property	Type	Necessity	Description
encryptedRequestData	string	Yes	The encryrpted request data. The request data is described in payload.
IV	string	Yes	Initial Value applied in the encryption

Sample

```
{
  "encryptedRequestData":
    "g+cQC8qGEDDd3lT7bG/S31PMKzSoHHnIsqv3QE9Ud8ZlVoAnmRLK6MPDPyEkzMLVtDXSY/ANYRKrew+u5WKbtzWorlRXGSkYh0+t37dd452l
    +NV33u4eFywGNDndybaiaKoVXWxaZCL80agqAJSfJ7t1EYYhnJE4UrauFLHY6s1K2L/75a2Sfcv0qH/VH2xI459nfIO4cHWrtUEbClmMwvSB20
    aKrI6pdchDwZZPvcgdn8usKGwy8eDVIxVzVbsJAcJ1MyF3TZmYAG66QauLv1/InNsPxqQbLeCCsBiY5Ggr5BslGdEECPJDVXcb7WAtJQrERMxj
    I/i6u3Mn4xb0qKjYq2MSIPpRvOg8qP7ay/10eRt2hwCkyfknoqM5cIstVfq4b0zGsZDJ8hx9m596DixLqDv9Q7AoooTOnbavrIB7bnOfMJlRvd
    HCK2vYB3yHoCrQUU=",
  "IV": "EIxcti+40UPSYH4L"
}
```

Payload

Property	Type(Max Len)	Necessity	Description
requestId	string(200)	Yes	Unique request id generated by vani
membershipId	string(10)	Yes	Membership id generated by vani
mobilePhoneNo	string(20)	Yes	Mobile phone number
dateOfBirth	string(10)	Yes	Customer’s birthday. format: YYYY-MM-DD ( <u>ISO-8601</u> (https://www.w3.org/TR/NOTE-datetime) )
gender	string(20)	Yes	Gender. M : Male, F : Female
customerName	string(512)	Yes	Customer name (Full Name)
emailAddress	string(256)	No	Email Address
vanilaBarcode	string(16)	Yes	vani’s barcode number. 16 dit
addressDetailInfo	object	No	Detail address information
state	string(500)	No	State (tỉnh / thành phố)
district	string(500)	No	District (Quận / Huyện)
detail1	string(500)	No	Detail address part1 면/동 (Phường/Xã)

Property	Type(Max Len)	Necessity	Description
detail2	string(500)	No	Detail address part2
countryCode	string(2)	No	countryCode defined at <a href="https://en.wikipedia.org/wiki/ISO_3166-1_alpha-2">ISO 3166-1 alpha-2</a> ( <a href="https://en.wikipedia.org/wiki/ISO_3166-1_alpha-2#Officially_assigned_code_elements">https://en.wikipedia.org/wiki/ISO_3166-1_alpha-2#Officially_assigned_code_elements</a> ) ex) VN , KR
idType	string(10)	No	Identification Type CMND : old ID card of Vietnam, CCCD , New ID card of Vietnam, PASSPORT : Passport
idNumber	string(20)	No	Identification Number for corresponding to the idType

### Response Body

Property	Type	Always Exists	Description
customerId	string	Yes	A Unique identifier of a customer (generated by Membership System)
membershipLevel	string	Yes	Membership Level (ex. VIP , GOLD ). NONE if level doesn't exist.
membershipPoint	number	Yes	Membership Point of the customer

### Error Code

If a problem occurs during the registration process, the membership system should provide error information defined by its own based on Error Response guide. However, for some situations defined below, the exact error code defined by vani should be delivered .

[VN]Nếu có lỗi xảy ra trong quá trình đăng ký, Membership System có thể cung cấp thông tin lỗi do chính hệ thống quy định theo tiêu chuẩn Error Response. Tuy nhiên, đối với một số tình huống được miêu tả dưới đây, thông tin lỗi phải được gửi dưới dạng error code do vani quy định.

[KO]Registration 과정에서 문제가 발생할 경우 멤버십 시스템은 Error Response 규격에 맞게 자체적으로 정의한 에러 정보를 전달할 수 있습니다. 단, 아래 정의하는 몇 가지 상황에 대해서는 반드시 vani가 정의한 error code 로 전달되어야 합니다.

Error Code	Reason	Impact on vani
DUP_EMAIL	<p>A customer failed to sign up for the membership because the customer's email already exists in the membership system.</p> <p>[VN]Khách hàng không đăng ký được thành viên vì email của khách hàng đã tồn tại trong hệ thống Membership System.</p> <p>[KO]가입과정에서 전달된 이메일이 이미 membership system에 존재하여 membership 회원가입이 실패 하였습니다.</p>	<p>vani Application displays appropriate information</p> <p>[VN]Ứng dụng vani hiển thị thông tin thích hợp.</p> <p>[KO]vani App에서는 적절한 정보를 고객에게 전달합니다.</p>

### Error Response Example

```
{
  "errorCode": "DUP_EMAIL",
  "errorMessage": "The email address already exists"
}
```



**customerId** **MUST NOT** be one of the user's information, such as mobile phone number or vanilaBarcode.

[VN] **customerId** không thể là thông tin cá nhân (Số điện thoại hoặc Mã vạch Vani)

[KO] **customerId** 는 고객정보 (휴대폰 번호나 Vanila 바코드)가 되어서는 안됩니다.

## Notifying vani user's information changes

vani may send any change of Customer's information to the Membership System.

[VN] Nếu thông tin của khách hàng thay đổi, vani có thể chuyển thông tin này đến Membership System.

[KO] vani는 고객의 정보가 변경될 경우 이 정보를 Membership System에 전달할 수 있습니다.

In order to receive the changed customer information provided by vani, Membership System should develop the API below and provide the URL to vani in advance.

[VN] Để nhận được thông tin khách hàng đã thay đổi do vani cung cấp, Membership System phải phát triển API bên dưới và cung cấp trước đường dẫn URL của Membership cho vani.

[KO] Membership System은 vani가 제공하는 변경된 고객정보를 받기 위해서는 아래 API를 개발하고 Membership의 URL 경로를 vani에 사전에 제공해야 합니다.



The Membership system must ensure that this API is idempotent. In other words, vani System can call this API multiple times when it does not receive a response due to network disconnection. In this case, Membership System should always deliver the same result

[VN] Membership System phải đảm bảo API này idempotent. Nói cách khác, vani System có thể gọi API này nhiều lần khi không nhận được phản hồi bình thường do bị ngắt kết nối mạng, v.v. Trong trường hợp này, Membership System phải luôn truyền đi cùng một kết quả.

[KO] Membership system은 이 API의 멍등성을 보장해야 합니다. 즉, vani SYstem은 네트워크 연결끊김 등으로 정상적인 응답을 못 받았을 때 이 API를 여러번 호출 할 수 있습니다. 이런 경우, Membership System은 항상 동일한 결과를 전달해야 합니다.

## Endpoint

POST /user-info-renewal

## JSON Structure

### Request Body

Property	Type	Necessity	Description
requestId	string	Yes	Unique request id generated by vani
customerId	string	Yes	A Unique identifier of a customer (generated by Membership System)
customerName	string	Yes	Customer Name
emailAddress	string	No	Email Address
addressDetailInfo	object	No	Detail address information

Property	Type	Necessity	Description
state	string	No	State (tỉnh / thành phố)
district	string	No	District (Quận / Huyện)
detail1	string	No	Detail address part1 (Phường/Xã)
detail2	string	No	Detail address part2

### Response Body

None

## Searching Membership Information

vani searches the details of the membership information of a customer

[VN]vani tra cứu thông tin thành viên của khách hàng.

[KO]고객이 보유한 Membership의 세부 정보를 조회 합니다.

### Endpoint

POST /detail-info-search

### JSON Structure

#### Request Body

Property	Type	Necessity	Description
customerId	string	Yes	A Unique identifier of a customer (generated by Membership System)

### Response Body

Property	Type	Always Exists	Description
availablePoint	number	Yes	available point of the customer. ( set to 0 if there is no or minus point.) [VN]Số điểm mà khách hàng có thể sử dụng [KO]고객이 사용할 수 있는 포인트
membershipLevel	string	Yes	current membership Level of the customer (ex. VIP , GOLD ). NONE if level doesn't exist. [VN]Hạng thành viên hiện tại của khách hàng (ex. VIP , GOLD ). NONE : Khi không tồn tại [KO]고객의 현재 멤버십레벨(ex. VIP , GOLD ). NONE : 존재하지 않을 경우

## Point history search

vani system inquires most recent points transaction of a customer.

[VN]vani tra cứu giao dịch điểm gần nhất của khách hàng

[KO]고객이 보유한 Membership 포인트의 가장 최근 거래를 조회합니다.

Endpoint

POST /points-history-details

JSON Structure

Request Body

Property	Type(Max Len)	Necessity	Description
customerId	string	Yes	A Unique identifier of a customer (generated by Membership System)
limitDataCount	number(5)	Yes	limit number of rows for the search result (maximum number: 100) [VN]Giới hạn số hàng cho kết quả tìm kiếm (số lượng tối đa: 100) [KO]포인트 이력 조회결과로 전달되는 data의 최대 건수, (최대숫자: 100)

Response Body

Property	Type	Always Exists	Description
pointHistoryList	array	Yes	Point History List
transactionId	string	Yes	Transaction id of point transaction [VN]Mã định danh duy nhất cho giao dịch điểm [KO]포인트 거래를 식별할 수 있는 고유 Id
pointType	string	Yes	Point Type ( EARN : points earn, REDEEM : points usage, CANCEL_EARN : points earn cancellation, CANCEL_REDEEM : points redeem cancellation, EXPIRY : points expiration) [VN]Loại điểm( EARN : Tích điểm, REDEEM : Sử dụng điểm, CANCEL_EARN : Hủy tích điểm , CANCEL_REDEEM : Hủy sử dụng điểm , EXPIRY : Hết hạn) [KO]포인트 타입 ( EARN : 적립, REDEEM : 사용, CANCEL_EARN : 포인트 적립취소, CANCEL_REDEEM : 포인트 사용취소, EXPIRY : 소멸)
transactionTime	string	Yes	The time of Point Transaction format: YYYY-MM-DDThh:mm:ssTZD ( <u>ISO-8601</u> ( <a href="https://www.w3.org/TR/NOTE-datetime">https://www.w3.org/TR/NOTE-datetime</a> ) ) Example: 2017-07-17T13:42:40+07:00
where	string	Yes	The place where the point transaction occurred ( ex.Offline Shop’s branch name or online site name) [VN]Nơi tích hoặc sử dụng điểm (VD: tên cửa hàng hoặc tên website trực tuyến) [KO]포인트 적립 또는 사용 장소 (오프라인 매장이름이나 온라인 Site 이름)

Property	Type	Always Exists	Description
pointAmount	number	Yes	amount of points applied for this transaction. Always positive integer [VN]Số điểm được áp dụng cho giao dịch này. Điểm luôn là số nguyên dương [KO]이 포인트 거래에 적용된 포인트 금액, 항상 양의 정수

## vani disconnection

vani System provides information when a vani user deletes his/her own account or just disconnect the membership

[VN]vani System sẽ gửi thông tin khi người dùng vani xóa tài khoản của mình hoặc hủy kết nối tư cách thành viên.

[KO]vani 가입자가 탈퇴하거나 단순히 멤버십 연결을 끊을 경우 정보를 전달합니다.



Membership System should delete or deactivate the vani barcode number information.

[VN]Membership System cần xóa hoặc vô hiệu hoá thông tin mã vạch của vani.

[KO]Membership System은 보유한 vani barcode 번호 정보등을 삭제 하거나 비활성화 해야 합니다.



If a user of vani App sign off and sign in again, vani System creates a new barcode. However, if a user simply disconnect the connection between vani and a Membership and reconnect, the existing barcode is delivered as it is.

[VN]Khi người dùng xóa tài khoản tại vani System và đăng ký thành viên lại, vani system sẽ tạo một mã vạch mới. Tuy nhiên, trong trường hợp chỉ hủy kết nối và kết nối lại với chương trình thành viên, mã vạch đã có sẵn sẽ được gửi đi như ban đầu.

[KO]vani system의 user가 탈퇴하고 재가입 하는 경우 vani System은 새로운 barcode를 생성합니다. 하지만 단순히 멤버십과의 연결을 끊고 다시 연결하는 경우는 기존 barcode를 그대로 전달합니다.



The Membership system must ensure that this API is idempotent. In other words, vani System can call this API multiple times when it does not receive a response due to network disconnection. In this case, Membership System should always deliver the same result

[VN]Membership System phải đảm bảo API này idempotent. Nói cách khác, vani System có thể gọi API này nhiều lần khi không nhận được phản hồi bình thường do bị ngắt kết nối mạng, v.v. Trong trường hợp này, Membership System phải luôn truyền đi cùng một kết quả.

[KO]Membership system은 이 API의 멍등성을 보장해야 합니다. 즉, vani SYstem은 네트워크 연결끊김 등으로 정상적인 응답을 못 받았을 때 이 API를 여러번 호출 할 수 있습니다. 이런경우 Membership System은 항상 동일한 결과를 전달해야 합니다.

## Endpoint

POST /disconnection

## JSON Structure

### Request Body

Property	Type(Max Len)	Necessity	Description
customerId	string	Yes	A Unique identifier of a customer (generated by Membership System)

vanilaBarcode	string(16)	Yes	vani's barcode number. 16 dit
reason	string(20)	Yes	<p>reason for disconnection</p> <p>[VN]Lý do yêu cầu huỷ kết nối</p> <p>[KO]disconnect를 요청한 사유</p> <p><b>SIGN-OFF</b> :a vani user sign off vani App</p> <p>[VN]Khi khách hàng của vani xoá tài khoản tại vani App</p> <p>[KO]vani고객이 vani App을 탈퇴할 경우</p> <p><b>DISCONNECT</b> : the vani user just disconnects the link between vani and the Membership in the vani App.</p> <p>[VN]Khi khách hàng của vani hủy kết nối với chương trình thành viên của công ty đối tác</p> <p>[KO]vani고객이 vani App에서 연결된 제휴사와의 연결을 끊을 경우</p>

### Response Body

None

## Issuing Coupon (Optional)

vani requests issuing a merchant's coupon only if a vani user meet the condition of a promotion.

[VN]vani yêu cầu Merchant phát hành coupon khi người dùng của vani đáp ứng đủ các điều kiện promotion.

[KO] vani는 프로모션 조건에 맞는 이벤트가 발생할 경우 Merchant에게 쿠폰발행을 요청합니다.



This API is Optional. A merchant may develop this API only if a merchant has a coupon system and allow vani system request issuing a coupon in real time. Usually, this function is more applicable for an online merchant because using the coupon in the merchant app is more convenient than using it in the vani app.

[VN]Đây không phải là API bắt buộc. Merchant chỉ có thể phát triển API này nếu có hệ thống coupon riêng biệt và có thể cho phép vani yêu cầu phát hành coupon theo thời gian thực. Thông thường, tính năng này chủ yếu được áp dụng cho các Online Merchant, vì người dùng sẽ dễ dàng nhận và sử dụng mã coupon tại app của Online Merchant.

[KO] 이 API 개발은 필수가 아닙니다. 자체 쿠폰 시스템을 가지고 있는 Merchant가 vani로부터 실시간으로 쿠폰 발급요청을 받아서 처리할 수 있는 경우, Merchant는 이 API를 개발할 수 있습니다. 일반적으로 이 기능은 주로 온라인 상점에서 활용되는데, 온라인 상점인 경우 사용자가 쿠폰을 온라인 상점 App에 발급받아 사용하는 것이 더 편리하기 때문입니다.



The Membership system must ensure that this API is idempotent. In other words, vani System can call this API multiple times when it does not receive a response due to network disconnection. In this case, Membership System should always deliver the same result

[VN]Membership System phải đảm bảo API này idempotent. Nói cách khác, vani System có thể gọi API này nhiều lần khi không nhận được phản hồi bình thường do bị ngắt kết nối mạng, v.v. Trong trường hợp này, Membership System phải luôn truyền đi cùng một kết quả.

[KO]Membership system은 이 API의 멍등성을 보장해야 합니다. 즉, vani system은 네트워크 연결끊김 등으로 정상적인 응답을 못 받았을 때 이 API를 여러번 호출할 수 있습니다. 이런 경우 Membership system은 항상 동일한 결과를 전달해야 합니다.

### Endpoint

POST /coupon-issuance

### JSON Structure

### Request Body



Property	Type(Max Len)	Necessity	Description
requestId	string(200)	Yes	Unique request id generated by vani
customerId	string	Yes	A Unique identifier of a customer (generated by Membership System)
vaniPromotionCode	string(128)	Yes	A promotion code generated by vani for each promotion [VN]Mã định danh từng promotion được tạo bởi vani [KO]vani에서 생성한 프로모션별 고유 코드

Response Body

Property	Type	Always Exists	Description
couponType	string	Yes	<p>The value should be <b>PRODUCT</b> or <b>TICKET</b></p> <p>[VN]Giá trị này phải là <b>PRODUCT</b> hoặc <b>TICKET</b> [KO]이 값은 <b>PRODUCT</b> 또는 <b>TICKET</b> 중 하나여야 합니다.</p> <p><b>PRODUCT</b> : this coupon is product coupon that does not have coupon number. [VN]Coupon này là coupon sản phẩm, không có mã coupon. [KO]이 쿠폰은 쿠폰 번호가 없는 상품쿠폰 입니다.</p> <p><b>TICKET</b> : this coupon is a coupon that has a coupon number. [VN]Coupon này là coupon có mã coupon. [KO]이 쿠폰은 쿠폰 번호가 있는 쿠폰 입니다.</p>
couponCode	string	Yes	<p>This code is unique by each promotion, not by each user.</p> <p>[VN]Mã coupon này là mã định danh từng promotion, không phải giá trị định danh theo người dùng. [KO]이 쿠폰 코드는 프로모션별로 유일한 값이고, 사용자별로 유일한 값 이 아닙니다.</p>
couponNumber	string	No	<p>Individual coupon number for each coupon. This number must be delivered only if the <b>couponType</b> is <b>TICKET</b></p> <p>[VN]Đây là mã coupon cá nhân. Mã chỉ được gửi khi <b>couponType</b> là <b>TICKET</b> . [KO]개별 쿠폰 번호 입니다. <b>couponCode</b> 가 <b>TICKET</b> 인 경우만 전달합니다.</p>
validFrom	string	YES	<p>The vaild starting day of a coupon (FORMAT: YYYY-MM-DD)</p> <p>[VN]Ngày bắt đầu thời hạn sử dụng coupon [KO]쿠폰의 유효시작일 (포맷: YYYY-MM-DD)</p>

validUntil	string	YES	The valid ending day of a coupon (FORMAT: YYYY-MM-DD) [VN]Ngày kết thúc thời hạn sử dụng coupon [KO]쿠폰의 유효종료일 (포맷: YYYY-MM-DD)
------------	--------	-----	--

## Checking Coupon Status (Optional)

It retrieves the current status of a coupon issuance requested by vani System.

[VN]Kiểm tra trạng thái coupon được vani yêu cầu

[KO]vani에서 요청한 쿠폰에 대한 상태를 확인 합니다.



This API is called when there is no response within 10 seconds to the coupon issuance request or business operator happened to check the coupon status by customer's request.

[VN]API này được gọi khi không nhận được phản hồi yêu cầu phát hành coupon sau 10s, hoặc khi kiểm tra trạng thái coupon theo yêu cầu của khách hàng.

[KO]vani System 에서 요청한 쿠폰발급 요청에 대해 10초 내에 응답이 없거나 고객의 요청에 의해 쿠폰 상태를 확인할 경우 이 API 를 호출합니다.

## Endpoint

GET /check-status-coupon/{requestId}

requestId is the unique requestId of /coupon-issuance API

[VN]requestId là mã nhận được khi yêu cầu API /coupon-issuance

[KO]requestId는 /coupon-issuance API 요청시 전달된 requestId 입니다.

## JSON Structure

### Request Body

None

### Response Body

Property	Type	Always Exists	Description
----------	------	---------------	-------------

Property	Type	Always Exists	Description
status	string	Yes	<p>The value should be one of ISSUED , USED , EXPIRED , CANCELED</p> <p>[VN]Giá trị này phải là ISSUED , USED , EXPIRED hoặc CANCELED .</p> <p>[KO]이 값은 ISSUED , USED , EXPIRED , CANCELED 중 하나여야 합니다.</p> <p><b>ISSUED</b> : a coupon has been issued but not used yet</p> <p>[VN]Trạng thái coupon đã được phát hành và chưa được sử dụng</p> <p>[KO]쿠폰이 발급되었고 아직 사용되지 않은 상태</p> <p><b>USED</b> : a coupon issued and already used</p> <p>[VN]Trạng thái coupon đã được phát hành và đã được sử dụng</p> <p>[KO]쿠폰이 발급된 이후 사용한 상태</p> <p><b>EXPIRED</b> : a coupon is expired</p> <p>[VN]Trạng thái coupon đã hết hạn sử dụng</p> <p>[KO]쿠폰의 유효기간이 만료된 상태</p> <p><b>CANCELED</b> :a coupon is canceled and unavailable</p> <p>[VN]Trạng thái coupon đã bị huỷ và không được sử dụng</p> <p>[KO]쿠폰이 취소되어 사용할 수 없는 상태</p>



If no matching data regarding the requestId, the HTTP Status Code should be passed as **404** .

[VN]Khi không có dữ liệu tương ứng với requestId, HTTP Status Code phải là 404.

[KO]requestId에 해당하는 데이터가 없는경우 HTTP Status Code는 **404** 로 전달되어야 합니다.

# REST API for Membership (Membership → vani)

This section describes the membership related API provided by vani System.

[VN]Hạng mục này mô tả API liên quan đến Membership được cung cấp bởi vani System.

[KO]vani System에서 제공하는 Membership과 관련된 API 입니다.

## Server Information

Base URL and API Version information of vani’s Server is as follows.

Environment	Base URL	API Version	description
Staging	https://membership-api.test.vani.la	v1.0	For testing and QA
Production	https://membership-api.vani.la	v1.0	For production Service

The API provided by vani System is called according to the API Versioning guide. For example, if the endpoint is `deregistration` , and to access the staging environment, you need to configure the following URL and call the vani System.

[VN]API do vani System cung cấp được gọi theo hướng dẫn API Versioning. Ví dụ: nếu endpoint là `deregistration` và để truy cập môi trường Staging, bạn cần định cấu hình URL như sau và gọi vani System.

[KO]vani가 제공하는 API는 API Versioning 가이드에 따라 호출합니다. 예를들어, endpoint가 `deregistration` 이고 Staging환경에 접속하기 위해서는 아래와 같은 URL을 구성해서 vani System을 호출해야 합니다.

<https://membership-test.vani.la/v1.0/deregistration> (https://membership-test.vani.la/v1.0/deregistration)

## Notifying Membership Deregistration

If a membership account is deleted, details should be sent to vani System.

[VN]Nếu tài khoản thành viên bị xoá, thông tin chi tiết cần được gửi đến vani System.

[KO]Membership이 해지 되었을 경우 vani로 내역을 전달합니다.

### Endpoint

POST /deregistration

### JSON Structure

#### Request Body

Property	Type	Necessity	Description
customerId	string	Yes	A Unique identifier of a customer (generated by Membership System)

#### Response Body

None

## Searching vani Barcode Number (Optional)

Merchant system can search for a valid vani barcode number based on the customer ID of the merchant system.

[VN]Merchant system có thể tra cứu số mã vạch vani hợp lệ dựa trên ID khách hàng của Merchant system.

[KO]Merchant system의 customer Id 를 기준으로 유효한 vani의 barcode number를 조회 할 수 있습니다.

### Endpoint

POST /barcode-search

### JSON Structure

#### Request Body

Property	Type	Necessity	Description
customerId	string	Yes	A Unique identifier of a customer (generated by Membership System)

#### Response Body

Property	Type(Max Len)	Always Exists	Description
vanilaBarcode	string(16)	Yes	vani's barcode number. 16 dit
amountOfVaniCoin	number(16)	Yes	the amount of vani coin(point). 0 or positive integer [VN]Giá trị của Vani Xu (point). Là 0 hoặc số nguyên dương. [KO] vani coin(point)의 값. 0 또는 양의 정수

## Delivering Membership Point Transaction Data (realtime)

The Membership System delivers relevant information to vani in real time when a point earning/redemption event occurs.

[VN]Membership System cung cấp thông tin liên quan cho vani trong thời gian thực khi có phát sinh giao dịch tích lũy / sử dụng điểm.

[KO]Membership System은 vani 가입 고객에 한해서 point 적립/사용 event가 발생할 경우 vani에게 실시간으로 관련 정보를 전달합니다.

Endpoint

POST /points-approval

JSON Structure

Request Body

Property	Type(Max Len)	Necessity	Description
transactionId	string	Yes	Unique transaction identifier for a point transaction
customerId	string	Yes	A Unique identifier of a customer (generated by Membership System)
isVanilaBarcodeUsed	boolean	Yes	<p>true if this transaction is done by vani barcode, false if not,</p> <p>true if this trnsaction is done by online Application,</p> <p>[VN] true nếu giao dịch này sử dụng mã vạch vani, nếu không thì là false</p> <p>Nếu là giao dịch online thì luôn là true</p> <p>[KO]바닐라 바코드를 사용한 거래면 true ,아니면 false ,</p> <p>온라인 거래인 경우 항상 true</p>
transactionType	string	Yes	<p>Transaction Type ( EARN : earning, REDEEM : redemption, DISCOUNT :discount )</p> <p>[VN]Loại giao dịch (EARN: tích điểm, REDEEM: sử dụng điểm ,DISCOUNT: nhận chiết khấu)</p> <p>[KO]거래 Type ( EARN : 적립, REDEEM : 사용, DISCOUNT :할인)</p>
transactionTime	string	Yes	<p>Transaction time of the point format: YYYY-MM-DDThh:mm:ssTZD ( <u>ISO-8601</u> )</p> <p>(<a href="https://www.w3.org/TR/NOTE-datetime">https://www.w3.org/TR/NOTE-datetime</a>) )</p> <p>Example: 2017-07-17T13:42:40+07:00</p>
brandId	string(5)	Yes	<p>Brand identifier defined by vani</p> <p>[VN]ID định danh thương hiệu quy định bởi vani System</p> <p>[KO]vani System이 정의한 brand 식별자</p>
shopBrandName	string(128)	Yes	brand Name of the merchant (ex. Pharmacity)
branch	string(128)	Yes	Branch Name of the brand (ex. Lý Chính Thắng )
paymentMethod	string	Yes	<p>The payment method accompanied by point transaction ( CASH , E-WALLET , CARD , ETC )</p> <p>[VN]Phương thức thanh toán đi kèm với giao dịch điểm</p> <p>[KO]Point 거래에 동반한 거래에 대한 결제수단</p> <p>Merchant should deliver appropriate paymentMethod</p> <p>[VN] Đối tác cần gửi phương thức thanh toán phù hợp cho từng giao dịch</p> <p>[KO]제휴사는 반드시 적절한 결제수단 정보를 제공해야 합니다.</p>

Property	Type(Max Len)	Necessity	Description
productTitle	string	Yes	A product title of a point transaction [VN]Tên sản phẩm đi kèm giao dịch điểm. Chỉ gửi một tên sản phẩm dù số lượng sản phẩm là 2 trở lên [KO]포인트 결제에 동반한 상품의 이름. 2개 이상일 경우 하나만 전달
productAmount	number	Yes	Original product amount of a point transaction before applying discount including tax [VN] Giá gốc sản phẩm đã bao gồm thuế, chưa tính giảm giá [KO]할인 적용 전, 세금포함 상품의 고유금액
billAmount	number	Yes	Actual bill amount of a point transaction including tax that a customer pays excluding discount amount (Coupon, Voucher, etc) [VN] Số tiền khách hàng cần phải trả (đã bao gồm thuế) sau khi trừ đi số tiền được giảm (áp dụng Coupon, Voucher) trên giá gốc sản phẩm [KO]상품금액에서 할인(Coupon, Voucher 등) 액을 제외하고 세금 포함 한 고객이 지불한 금액
productPurchaseTime	string	Yes	Product purchase time. format: YYYY-MM-DDThh:mm:ssTZD ( <u>ISO-8601</u> ) ( <a href="https://www.w3.org/TR/NOTE-datetime">https://www.w3.org/TR/NOTE-datetime</a> ) ) Example: 2017-07-17T13:42:40+07:00
productCurrency	string	Yes	Currency code of product amount ( ISO 4217, alphabetic code) Example: VND for Vietnamese đồng
pointAmount	number	Yes	Amount of points for this point transaction. [VN]Số điểm tích hoặc sử dụng [KO]적립 또는 사용 포인트
totalPointAmount	number	Yes	Total remain amount of points after this point transaction [VN]Tổng số điểm hiện tại còn lại sau giao dịch điểm này [KO]적립 또는 포인트 사용후 남은 현재 포인트
discountAmount	number	No	Discount Amount. If transactionType is DISCOUNT , then this value should be sent. [VN]Số tiền chiết khấu. Nếu transactionType là DISCOUNT thì đây là giá trị bắt buộc [KO]할인금액. transactionType 이 DISCOUNT 인 경우 필수 값

Property	Type(Max Len)	Necessity	Description
trafficType	String	No (Yes if ONLINE)	<p>OFFLINE if this transaction occurred via offline shop, ONLINE if this transaction occurred via online application</p> <p>[VN]Khi phát sinh giao dịch tại cửa hàng offline thì "OFFLINE", phát sinh giao dịch tại Application online thì "ONLINE"</p> <p>[KO]Offline 매장에서 거래가 발생할 경우 OFFLINE , 온라인 어플리케이션에서 거래가 발생할 경우 ONLINE</p>
vaniCouponNumber	String	No (Yes if Vani's Coupon applied)	<p>Vani's coupon number. If a vani coupon is applied in the transaction, the coupon number should be delivered</p> <p>[VN]Mã coupon của vani. Mã coupon phải được truyền nếu coupon của vani được sử dụng trong giao dịch này.</p> <p>[KO]vani쿠폰번호, 만약 이 거래에 vani 쿠폰이 적용된 경우, 쿠폰번호를 전달해야 합니다.</p>



If more than one vani coupons are applied in one transaction, the **vaniCouponNumber** may include 2 or more coupon numbers by comma(,) delimiter.

[VN]Khi một giao dịch được áp dụng nhiều hơn một vani coupon, **vaniCouponNumber** có thể gồm 2 hoặc nhiều mã được phân cách bằng dấu phẩy (,).

[KO]만약 적용되는 vani coupon이 2개이상일 경우 **vaniCouponNumber** 는 콤마(,)로 구분해서 전달 할 수 있습니다.

### Response Body

None

### Error Code

Vani system delivers the appropriate HTTP Status Code (4XX) and error Information.

[VN]

[KO]전달된 data에 문제가 있는경우 기본적으로 HTTP Status Code 4XX를 전달하고 적절한 에러 정보를 전달합니다.



Error Code	Reason	Merchant's action
DUP_TRANSACTION_ID	<p>The transactionId of the requested data is identical with the previous transaction data.</p> <p>[VN]transactionId trùng với transactionId nằm trong data trước đó.</p> <p>[KO]transactionId가 이전 data에 포함된 transactioinId와 동일합니다.</p>	<p>The merchant should check duplication of the transactionId. Then the data should be fixed and delivered again if the transactionId is wrong.</p> <p>[VN]Đối tác cần kiểm tra lý do transactionId bị trùng. Khi transactionId bị trùng, đối tác cần chỉnh sửa và truyền lại data đúng</p> <p>[KO]제휴사는 동일한 transactionId가 전달된 사유를 확인해야 합니다. transactionId가 잘못되었다면 수정 후 다시 전달해야 합니다.</p>
DUP_TRANSACTION_DATA	<p>The transactionId is different from previously delivered data, but the other data is the same.</p> <p>[VN]transactionId khác nhưng bị trùng các data còn lại.</p> <p>[KO]transactionId는 다르지만 다른 데이터는 동일합니다.</p>	<p>The merchant should check duplication of the transaction data and fix it for not happening again.</p> <p>[VN]Đối tác cần kiểm tra lại lý do các data bị trùng tránh phát sinh lại lỗi tương tự</p> <p>[KO]제휴사는 동일한 data 가 전달된 사유를 확인하고 하고 재발하지 않도록 수정해야 합니다.</p>
UNAUTHORIZED	<p>The request is unauthorized.</p> <p>[VN]Yêu cầu không được phép</p> <p>[KO]허가되지 않은 요청입니다.</p>	<p>The merchant should check the API Key and fix it.</p> <p>[VN]Đối tác cần kiểm tra lại API Key và sửa lại nếu chưa đúng.</p> <p>[KO]제휴사는 API Key가 맞는지 확인하고 수정해야합니다.</p>
BAD_REQUEST	<p>The request does not follow API Spec.</p> <p>[VN]Yêu cầu chưa đúng chuẩn của API Spec.</p> <p>[KO]이 요청은 API Spec을 따르지 않고 있습니다.</p>	<p>The merchant should fix it. It might have typo or wrong type field</p> <p>[VN]Đối tác cần chỉnh sửa lại yêu cầu JSON data. Có thể do field bị sai chính tả hoặc sai kiểu.</p> <p>[KO]제휴사는 요청 JSON data를 수정해야 합니다. 오타자나 잘못된 type을 가진 field가 있을 수 있습니다.</p>

### Error Response Example

```
{
  "errorCode": "DUP_TRANSACTION_ID",
  "errorMessage": "the transactionId is duplicated. Please check the the request JSON."
}
```

## Delivering Membership Point Transaction Data (Bulk Insert)

The Membership System can send a number of point earning/redemption transactions.

[VN]Membership System có thể gửi đi thông tin nhiều giao dịch tích điểm/sử dụng điểm.

[KO]Membership System은 다수의 포인트 적립/사용 거래를 보낼 수 있습니다.



This API should be used only when the Membership System can not send point transaction data in real-time.

[VN]API này chỉ nên được sử dụng khi Membership System không thể gửi dữ liệu giao dịch điểm trong thời gian thực, mà phải gửi thông tin Point Transaction theo số lượng lớn tại một thời điểm nhất định.

[KO]이 API는 realtime 전달이 어려워 특정 시점에 대량으로 Point Transaction 정보를 전달해야 하는 경우에만 사용해야 합니다.



The Membership system can call this API multiple times with the same transaction id when a normal response is not received due to network disconnection. In this case, the vani System always delivers the same result.

[VN]Membership system có thể gọi API này nhiều lần bằng cùng một transaction id khi không nhận được phản hồi bình thường do bị ngắt kết nối mạng, v.v. Trong trường hợp này, vani System phải luôn truyền đi cùng một kết quả.

[KO]Membership system은 네트워크 연결끊김 등으로 정상적인 응답을 못 받았을 때 이 동일한 transaction id로 API를 여러번 호출 할 수 있습니다. 이런경우, vani System은 항상 동일한 결과를 전달합니다.

### Endpoint

POST /points-approval-bulk



The maximum number of data array are 100

[VN]Số dữ liệu tối đa array có thể chứa là 100.

[KO]array에 담을 수 있는 data의 개수는 최대 100개 이다.

### JSON Structure

#### Request Body

Property	Type	Necessity	Description
transactionDataList	array	Yes	Transaction data list
transactionId	string	Yes	Unique identifier for a point transaction
customerId	string	Yes	A Unique identifier of a customer (generated by Membership System)

isVanilaBarcodeUsed	boolean	Yes	<p><code>true</code> if this transaction is done by vani barcode, <code>false</code> if not,</p> <p><code>true</code> if this transaction is done by online Application</p> <p>[VN] <code>true</code> nếu giao dịch này sử dụng mã vạch vani, nếu không thì là <code>false</code></p> <p>Nếu là giao dịch online thì luôn là <code>true</code></p> <p>[KO]바닐라 바코드를 사용한 거래면 <code>true</code>,아니면 <code>false</code>, 온라인 거래일 경우 항상 <code>true</code></p>
transactionType	string	Yes	<p>Transaction Type ( <code>EARN</code> : earning, <code>REDEEM</code> : redemption, <code>DISCOUNT</code> :discount )</p> <p>[VN]Loại giao dịch (EARN: tích điểm, REDEEM: sử dụng điểm, DISCOUNT: nhận chiết khấu)</p> <p>[KO]거래 Type ( <code>EARN</code> : 적립, <code>REDEEM</code> : 사용, <code>DISCOUNT</code> :할인)</p>
transactionTime	string	Yes	<p>Transaction time. format: YYYY-MM-DDThh:mm:ssTZD ( <a href="https://www.w3.org/TR/NOTE-datetime">ISO-8601</a> (https://www.w3.org/TR/NOTE-datetime) )</p> <p>Example: 2017-07-17T13:42:40+01:00</p>
brandId	string(5)	Yes	<p>Brand identifier defined by vani</p> <p>[VN]ID nhận diện thương hiệu quy định bởi vani System</p> <p>[KO]vani System이 정의한 brand 식별자</p>
shopBrandName	string(128)	Yes	brand Name of the merchant (ex. Pharmacy)
branch	string(128)	Yes	Branch Name of the brand (ex. Lý Chính Thắng )
paymentMethod	string	Yes	<p>The payment method accompanied by point transaction ( <code>CASH</code> , <code>E-WALLET</code> , <code>CARD</code> , ETC )</p> <p>[VN]Phương thức thanh toán đi kèm với giao dịch tích điểm</p> <p>[KO]Point 거래에 동반한 거래에 대한 결제수단</p> <p>Merchant should deliver appropriate paymentMethod</p> <p>[VN] Đối tác cần gửi phương thức thanh toán phù hợp cho từng giao dịch</p> <p>[KO]제휴사는 반드시 적절한 결제수단 정보를 제공해야 합니다.</p>
productTitle	string	Yes	<p>A product title of a point transaction</p> <p>[VN]Tên sản phẩm đi kèm giao dịch điểm. Chỉ gửi một tên sản phẩm dù số lượng sản phẩm là 2 trở lên</p> <p>[KO]포인트 결제에 동반한 상품의 이름. 2개 이상일 경우 하나만 전달</p>
productAmount	number	Yes	<p>Original product amount of a point transaction before applying discount including tax</p> <p>[VN] Giá gốc sản phẩm đã bao gồm thuế, chưa tính giảm giá</p> <p>[KO]할인 적용 전, 세금포함 상품의 고유금액</p>

billAmount	number	Yes	<p>Actual bill amount of a point transaction including tax that a customer pays excluding discount amount (Coupon, Voucher, etc)</p> <p>[VN] Số tiền khách hàng cần phải trả (đã bao gồm thuế) sau khi trừ đi số tiền được giảm (áp dụng Coupon, Voucher) trên giá gốc sản phẩm</p> <p>[KO] 상품금액에서 할인(Coupon, Voucher 등) 액을 제외하고 세금 포함한 고객이 지불한 금액</p>
productPurchaseTime	string	Yes	<p>Product purchase time. format: YYYY-MM-DDThh:mm:ssTZD ( <a href="https://www.w3.org/TR/NOTE-datetime">ISO-8601</a> ) (<a href="https://www.w3.org/TR/NOTE-datetime">https://www.w3.org/TR/NOTE-datetime</a>) )</p> <p>Example: 2017-07-17T13:42:40+01:00</p>
productCurrency	string	Yes	<p>Currency code of product amount ( ISO 4217, alphabetic code)</p> <p>Example: VND for Vietnamese đồng</p>
pointAmount	number	Yes	<p>Amount of points for this point transaction.+ [VN]Số điểm tích hoặc sử dụng.</p> <p>[KO]적립 또는 사용 포인트</p>
totalPointAmount	number	Yes	<p>Total remain amount of points after this point transaction</p> <p>[VN]Tổng số điểm hiện tại còn lại sau giao dịch điểm này</p> <p>[KO]적립 또는 포인트 사용후 남은 현재 포인트</p>
discountAmount	number	No	<p>Discount Amount. If transactionType is DISCOUNT , then this value should be sent.</p> <p>[VN]Số tiền chiết khấu. Nếu transactionType là DISCOUNT thì đây là giá trị bắt buộc</p> <p>[KO]할인금액. transactionType 이 DISCOUNT 인 경우 필수 값</p>
trafficType	String	No (Yes if ONLINE)	<p>OFFLINE if this transaction occurred via offline shop, ONLINE if this transaction occurred via online application</p> <p>[VN]Khi phát sinh giao dịch tại cửa hàng offline thì "OFFLINE", phát sinh giao dịch tại Application online thì "ONLINE"</p> <p>[KO]Offline 매장에서 거래가 발생할 경우 OFFLINE , 온라인 어플리케이션에서 거래가 발생할 경우 ONLINE</p>
vaniCouponNumber	String	No (Yes if Vani's Coupon applied)	<p>Vani's coupon number. If a vani coupon is applied in the transaction, the coupon number should be delivered</p> <p>[VN]Mã coupon của vani. Mã coupon phải được truyền nếu coupon của vani được sử dụng trong giao dịch này.</p> <p>[KO]vani쿠폰번호, 만약 이 거래에 vani 쿠폰이 적용된 경우, 쿠폰번호를 전달해야 합니다.</p>



If more than one vani coupons are applied in one transaction, the `vaniCouponNumber` may include 2 or more coupon numbers by comma(,) delimiter.

[VN]Khi một giao dịch được áp dụng nhiều hơn một vani coupon, `vaniCouponNumber` có thể gồm 2 hoặc nhiều mã được phân cách bằng dấu phẩy (,).

[KO]만약 적용되는 vani coupon이 2개이상일 경우 `vaniCouponNumber` 는 콤마(,)로 구분해서 전달 할 수 있습니다.

## Response Body

None

## Delivering Membership Point Cancellation data

The Membership System should deliver the information to the vani System in real time when the point earning/redemption transaction is cancelled.

[KO]Membership System은 point 적립/사용 거래가 취소될 때 그 정보를 실시간으로 vani System에 전달해야 합니다.



The Membership system can call this API multiple times with the same transaction id when a normal response is not received due to network disconnection. In this case, the vani System always delivers the same result.

[VN]Membership system có thể gọi API này nhiều lần bằng cùng một transaction id khi không nhận được phản hồi bình thường do bị ngắt kết nối mạng, v.v. Trong trường hợp này, vani System phải luôn truyền đi cùng một kết quả.

[KO]Membership system은 네트워크 연결끊김 등으로 정상적인 응답을 못 받았을 때 이 동일한 transaction id로 API를 여러번 호출 할 수 있습니다. 이런경우, vani System은 항상 동일한 결과를 전달합니다.

## Endpoint

POST /points-cancellation

## JSON Structure

### Request Body

Property	Type	Necessity	Description
transactionId	string	Yes	Unique identifier for a cancellation transaction
customerId	string	Yes	A Unique identifier of a customer (generated by Membership System)
orgTransactionId	string	Yes	<p>The target transaction id of this cancellation request [VN]ID giao dịch gốc (ID giao dịch mục tiêu của yêu cầu hủy này) [KO]원 거래 고유번호</p> <p>orgTransactionId should be identical with the transactionId of points-approval. [VN] Phải giống với transactionId của API points-approval. [KO]points-approval에 전달 되었던 transactionId와 동일해야 합니다.</p>
transactionTime	string	Yes	<p>Transaction time. format: YYYY-MM-DDThh:mm:ssTZD ( <a href="https://www.w3.org/TR/NOTE-datetime">ISO-8601</a> (https://www.w3.org/TR/NOTE-datetime) ) Example: 2017-07-17T13:42:40+01:00</p>
cancelType	string	Yes	Cancellation type. P : Partial Cancellation, T : Total Cancellation

Property	Type	Necessity	Description
cancelPointAmount	number	Yes	The amount of points for this cancellation.
totalPointAmount	number	No	Total remain amount of points after this cancellation transaction [VN]Tổng số điểm hiện tại còn lại sau giao dịch hủy này [KO]취소 처리 후 남은 현재 포인트

### Response Body

None

## Delivering Membership Point Cancellation data(Bulk Insert)

The Membership System can send a number of point cancellation transactions.

[VN]Hệ thống membership có thể gửi giao dịch hủy số điểm

[KO]Membership System은 다수의 포인트 취소 거래를 보낼 수 있습니다.



This API should be used only when the Membership System can not send point transaction data in real-time.

[VN]API này chỉ nên được sử dụng khi Membership System không thể gửi dữ liệu giao dịch điểm trong thời gian thực, mà phải gửi thông tin Point Transaction theo số lượng lớn tại một thời điểm nhất định.

[KO]이 API는 realtime 전달이 어려워 특정 시점에 대량으로 Point Transaction 정보를 전달해야 하는 경우에만 사용해야 합니다.



The Membership system can call this API multiple times with the same transaction id when a normal response is not received due to network disconnection. In this case, the vani System always delivers the same result.

[VN]Membership system có thể gọi API này nhiều lần bằng cùng một transaction id khi không nhận được phản hồi bình thường do bị ngắt kết nối mạng, v.v. Trong trường hợp này, vani System phải luôn truyền đi cùng một kết quả.

[KO]Membership system은 네트워크 연결끊김 등으로 정상적인 응답을 못 받았을 때 이 동일한 transaction id로 API를 여러번 호출 할 수 있습니다. 이런경우, vani System은 항상 동일한 결과를 전달합니다.

### Endpoint

POST /points-cancellation-bulk

### JSON Structure

#### Request Body

Property	Type	Necessity	Description
transactionDataList	array	Yes	Transaction data list
transactionId	string	Yes	Unique identifier for a cancellation transaction
customerId	string	Yes	A Unique identifier of a customer (generated by Membership System)

orgTransactionId	string	Yes	<p>The target transaction id of this cancellation request</p> <p>[VN]ID giao dịch gốc (ID giao dịch mục tiêu của yêu cầu hủy này)</p> <p>[KO]원 거래 고유번호</p> <p>orgTransactionId should be identical with the transactionId of points-approval.</p> <p>[VN] Phải giống với transactionId của API points-approval.</p> <p>[KO]points-approval에 전달 되었던 transactionId와 동일해야 합니다.</p>
transactionTime	string	Yes	<p>Transaction time of the cancellation format: YYYY-MM-DDThh:mm:ssTZD ( <u>ISO-8601</u> (https://www.w3.org/TR/NOTE-datetime) )</p> <p>Example: 2017-07-17T13:42:40+07:00</p>
cancelType	string	Yes	<p>Cancellation type. P : Partial Cancellation, T : Total Cancellation</p>
cancelPointAmount	number	Yes	<p>The amount of points for this cancellation.</p>
totalPointAmount	number	No	<p>Total remain amount of points after this cancellation transaction</p> <p>[VN]Tổng số điểm hiện tại còn lại sau giao dịch hủy này</p> <p>[KO]취소 처리 후 남은 현재 포인트</p>

### Response Body

None

## REST API for Point Swap (vani → Membership )

This section describes the API that the **Membership System should provide for point swap** with the vani System.

[VN]Mục này mô tả API mà Membership System cần cung cấp để hoán đổi điểm với vani System.

[KO]Membership System이 vani System과의 Point swap을 위해 제공해야 되는 API에 대해 설명합니다.

- Merchant's Membership System should comply with API Guideline for Membership System defined in this documentation.

### Checking Remain Amount of Points

vani System checks the remaining points currently held by customers in the Membership System.

[VN]vani System tra cứu số điểm còn lại mà khách hàng đang có trong Membership System.

[KO]vani System은 Membership System에 현재 고객이 가지고 있는 잔여 포인트를 조회합니다.

#### Endpoint

POST /points-seeking

#### JSON Structure

##### Request Body

customerId	string	Yes	A Unique identifier of a customer (generated by Membership System)
------------	--------	-----	--

##### Response Body

Property	Type	Always Exists	Description
totalPoints	number	Yes	The customer's remain total points [VN]Tổng số điểm đang có [KO]총 보유 포인트
swappablePoints	number	Yes	The amount of points that actually can be exchangeable [VN]Số điểm có thể đổi trên tổng số điểm [KO]총 Point 중 교환 가능한 point



**swappablePoints** is only meaningful if Membership Points are convertible to vani Points. If not , swappablePoints are set to 0.

[VN] **swappablePoints** chỉ có ý nghĩa khi Membership Point có thể chuyển đổi thành vani Point. Nếu không, swappablePoints được đặt thành 0.

[KO] **swappablePoints** 는 Membership point가 vani Point로 전환 가능한 경우에만 의미가 있다. 만약 전환가능하지 않으면 swappablePoints는 0 으로 설정 한다.

### Membership Points Refill

It deducts vani Points and recharges them with Membership Points.

[VN]Đây là tính năng trừ điểm vani Point và nạp điểm cho Membership Point.

[KO]vani point를 차감하고 membership의 point로 충전하는 기능입니다.

#### Endpoint



POST /points-refill

## JSON Structure

### Request Body

Property	Type(MaxLen)	Necessity	Description
requestId	string(200)	Yes	Unique request id generated by vani
customerId	string	Yes	A Unique identifier of a customer (generated by Membership System)
refillPoints	number(16)	Yes	The refilled amount of Membership Points that refill. (positive integer) [VN]Số Membership Point sẽ nạp( 1 số dương) [KO]충전할 Membership Point(양의 정수)
exchangedVanilaPoints	number(16)	Yes	The deducted amount of vani Points (positive integer) [VN]Số vani Point sẽ bị trừ( 1 số dương) [KO]차감된 vani Point (양의 정수)

### Response Body

Property	Type	Always Exists	Description
transactionId	string	Yes	Unique transaction id
transactionTime	string	Yes	Transaction time. format: YYYY-MM-DDThh:mm:ssTZD ( <a href="https://www.w3.org/TR/NOTE-datetime">ISO-8601</a> (https://www.w3.org/TR/NOTE-datetime) ) Example: 2017-07-17T13:42:40+01:00
totalPointAmount	number	Yes	Total remain amount of points after this point transaction [KO]처리 후 남은 현재 포인트 [VN]Tổng số điểm còn lại sau khi xử lý yêu cầu này

## Membership Point Usage

vani System makes a request for redemption of Merchant Points to recharge vani Points.

[VN]vani System yêu cầu sử dụng Merchant point để nạp vani Point.

[KO]vani System은 vani Point를 충전하기 위해 Merchant point에 대한 사용 요청을 합니다.



Membership may not offer this feature depending on your contract with vani.

[VN]Membership có thể không cung cấp tính năng này tùy thuộc vào hợp đồng đã thỏa thuận với vani

[KO]Membership은 vani와의 계약에 따라 이 기능을 제공 하지 않을 수 있습니다.

## Endpoint

POST /points-usage

## JSON Structure

### Request Body

Property	Type(Max Len)	Necessity	Description
requestId	string(200)	Yes	unique request id generated by vani
customerId	string	Yes	A Unique identifier of a customer (generated by Membership System)
usagePoints	number(16)	Yes	The amount of membership points for usage (positive integer) [VN]Số điểm thành viên (membership point) yêu cầu sử dụng (1 số dương) [KO]사용요청 제휴사포인트(양의정수)

### Response Body

Property	Type	Always Exists	Description
transactionId	string	Yes	Transaction id for the points usage
transactionTime	string	Yes	Transaction time. format: YYYY-MM-DDThh:mm:ssTZD ( <a href="https://www.w3.org/TR/NOTE-datetime">ISO-8601</a> (https://www.w3.org/TR/NOTE-datetime) ) Example: 2017-07-17T13:42:40+01:00
totalPointAmount	number	Yes	Total remain amount of points after this point transaction [VN]Số điểm còn lại sau khi xử lý yêu cầu này [KO]처리 후 남은 현재 포인트

## Cancellation of Membership Point refill or Usage

It requests cancellation of a previous point refill or usage transaction request.

[VN]Yêu cầu hủy nạp điểm hoặc sử dụng điểm.

[KO]Point 충전 또는 사용 요청에 대한 취소를 요청합니다.



Cancellation requests can be made by the customer through the vani Customer Center.

[VN]Khách hàng có thể yêu cầu hủy thông qua Trung tâm CSKH của vani

[KO]vani 고객 센터를 통해 고객이 취소 요청을 하는 경우 요청 할 수 있습니다.

vani System requests a full cancellation of the refilled or used amount of points. Membership System may reject this request if the customer already has used the refilled amount and does not have enough points.

[VN]vani System sẽ yêu cầu hủy toàn bộ số điểm đã nạp hoặc sử dụng. Membership System có thể từ chối yêu cầu này nếu khách hàng đã sử dụng số điểm được nạp và không có đủ điểm.

[KO]vani System은 충전 요청한 금액 전체를 취소 요청합니다. 만약 고객이 충전된 금액을 일부 사용했다면, Membership System은 충전 취소 요청에 대해 거절 할 수 있습니다.

### Endpoint

POST /points-cancellation

## JSON Structure

### Request Body

Property	Type(Max Len)	Necessity	Description
requestId	string(200)	Yes	Unique request id generated by vani
orgRequestId	string(200)	Yes	The request id of the previous refill or usage request [VN]Request ID của yêu cầu nạp hoặc sử dụng điểm trước đó [KO]충전 요청시 vani가 전달한 충전요청 고유번호
cancellationPoint	number(16)	Yes	The cancellation amount of points. The amount of point is identical of the previous requested refill or usage amount of points. [VN]Số điểm yêu cầu hủy. Gửi đi số điểm tương tự với số điểm đã yêu cầu nạp hoặc sử dụng. [KO]취소 요청하는 Point. 충전 또는 사용 요청했던 금액과 동일한 금액을 전달합니다.

### Response Body

Property	Type	Always Exists	Description
transactionId	String	Yes	Unique transaction id
transactionTime	string	Yes	Transaction time. format: YYYY-MM-DDThh:mm:ssTZD ( <a href="https://www.w3.org/TR/NOTE-datetime">ISO-8601</a> (https://www.w3.org/TR/NOTE-datetime) ) Example: 2017-07-17T13:42:40+01:00

## Transaction Information of Membership Point

It retrieves the result of a point transaction (refill/usage/cancellation) requested by vani System.

[VN]Tra cứu kết quả giao dịch nạp/sử dụng/hủy điểm mà vani System yêu cầu.

[KO]vani에서 요청한 포인트 관련 충전/사용/취소 대한 결과를 조회합니다.



If there is no response within 10 seconds to the point refill/usage/cancellation request, vani System checks whether vani's request has been processed through this API.

[VN]Nếu không có phản hồi trong vòng 10 giây đối với yêu cầu nạp/sử dụng/hủy điểm, vani System sẽ kiểm tra xem yêu cầu của vani đã được xử lý thông qua API này hay chưa.

[KO]vani System 에서 요청한 point 충전/사용/취소 요청에 대해 10초 내에 응답이 없을 경우, 이 API를 통해 vani의 요청에 대한 처리가 완료 되었는지를 확인 합니다.

### Endpoint

GET /points/{requestId}

requestId is the unique request identifier sent by vani System when requesting to refill/usage/cancellation points transaction.

[VN]requestId là mã yêu cầu định danh độc nhất do vani System gửi đi khi yêu cầu nạp / sử dụng / hủy điểm.

[KO]requestId는 포인트 충전/사용/취소 요청 시 vani에서 전달한 요청 고유 번호 입니다.

## JSON Structure

### *Request Body*

None

### *Response Body*

Property	Type	Always Exists	Description
status	string	Yes	Status of the transaction related to the requestId SUCCESS : successful, FAILURE : failure, PROCESSING : under processing and not finished yet, UNKNOWN_REQUEST_ID :unknown
transactionId	string	No	Membership system's transactionId related to the request id. If the status is UNKNOWN_REQUEST_ID , this field can be null
transactionTime	string	No	Transaction time of the request was completed. format: YYYY-MM-DDThh:mm:ssTZD ( <a href="https://www.w3.org/TR/NOTE-datetime">ISO-8601</a> ( <a href="https://www.w3.org/TR/NOTE-datetime">https://www.w3.org/TR/NOTE-datetime</a> ) ) Example: 2017-07-17T13:42:40+01:00 If the status is PROCESSING this field can be null

## Web page link (Only for Online Merchant)

vani provides a mobile web page so that merchant mobile web or application can install or call the vani application (Android, iOS) by calling the vani's web page in a browser.

[VN]vani cung cấp mobile web page để gọi hoặc cài đặt vani application (Android, iOS) tại web page hoặc mobile application của merchant

[KO]vani는 merchant의 mobile application 이나 web page에서 vani application (Android,iOS)을 설치하거나 호출 할 수 있도록 mobile web page를 제공합니다.

A merchant's mobile application or web page should check whether the user is a vani user connected to the merchant. If the user is not the vani user of the merchant, the merchant's mobile applicatioin or web page can call a browser with the url of the vani web page in order to let the user connect to vani application.

[VN]ại web page hoặc moblie application của merchant, kiểm tra xem khách hàng đó có phải là người dùng vani đã liên kết với merchant tương ứng hay không từ merchant server, nếu người dùng vẫn chưa liên kết thì ta có thể sử dụng browser để gọi web page mà vani cung cấp.

[KO]merchant의 mobile application 이나 web page에서는 고객이 해당 merchant와 연결된 vani user인지를 merchant server로 부터 확인하고연결이 안되어 있는 경우 vani가 제공하는 web page를 browser를 이용하여 호출할 수 있습니다.



For proper operation of the page provided by vani, it is recommended to call it from the browser, not the webview.

[VN]Để các thao tác của trang được vani cung cấp diễn ra thuận lợi, nên gọi tại browser chứ không phải webview.

[KO]vani에서 제공하는 page의 적절한 동작을 위해 webview가 아닌 browser에서 호출 하는것을 권고합니다.

## REST API for Filling Vani Coin(point) (Merchant → vani)

Merchant can make a filling request to vani by prior contract with vani. For example, a merchant can use this function when the merchant wants to give benefits to a specific customer with vani coin. Another example is that if a merchant has a point system, and the merchant want to do a point swap with vani coin at the Merchant App's side, it can also be used to request a filling of vani coin. In this case, vani only allows filling vani coin, not using vani coin.

[VN]Dựa vào hợp đồng được ký kết trước đó với vani, Merchant có thể yêu cầu nạp xu đến vani. Ví dụ, Merchant có thể sử dụng tính năng này khi muốn cấp ưu đãi là Vani Xu cho khách hàng. Hoặc nếu Merchant có point system, Merchant có thể sử dụng tính năng này yêu cầu trừ Merchant point và nạp Vani Xu khi muốn đổi Vani Xu tại Merchant App. Trong trường hợp này, chỉ hỗ trợ nạp Vani Xu, không hỗ trợ sử dụng Vani Xu.

[KO]Merchant는 vani와 사전 계약에 의해 Merchant는 vani로 충전 요청을 할 수 있습니다. 예를들면, Merchant가 특정 고객에게 혜택을 vain point로 주고 싶을 때 이 기능을 이용할 수 있습니다. 또 다른 예로, Merchant가 point system을 가진 경우 Merchant App 내에서 vani coin으로 point swap을 하고 싶을 경우에도 Merchant point를 차감하고 vani coin을 충전 요청할 때도 사용할 수 있습니다. 이 경우 vani는 충전만 지원하고, 사용은 지원하지 않습니다.

## Server Information

The Base URL and API Version information of vani's Server for filling vani coin is as follows.

Environment	Base URL	API Version	description
Staging	<a href="https://balance-api.test.vani.la">https://balance-api.test.vani.la</a>	v1.0	For testing and QA
Production	<a href="https://balance-api.vani.la">https://balance-api.vani.la</a>	v1.0	For production Service

The API provided by vani System is called according to the API Versioning guide. For example, if the endpoint is `points-filling`, and to access the staging environment, you need to configure the following URL and call the vani System.

[VN]API do vani System cung cấp được gọi theo hướng dẫn API Versioning. Ví dụ: nếu endpoint là `points-filling` và để truy cập môi trường Staging, bạn cần định cấu hình URL như sau và gọi vani System.

[KO]vani가 제공하는 API는 API Versioning 가이드에 따라 호출합니다. 예를들어, endpoint가 `points-filling` 이고 Staging환경에 접속하기 위해서는 아래와 같은 URL을 구성해서 vani System을 호출해야 합니다.

```
https://balance-api.test.vani.la/v1.0/points-filling
```

## Technical considerations

API related to point filling transaction should perform signature processing using JSON Web Token (<https://jwt.io>) for data delivered to the request body to enhance security.

[VN]Về API liên quan đến transation nạp xu, vì vấn đề bảo mật, data được truyền đến bằng Request Body phải sử dụng JSON Web Token (<https://jwt.io>) và xử lý Signature.

[KO]포인트 충전 Transaction 과 관련된 API는 보안 강화를 위해 Request Body로 전달되는 데이터를 JSON Web Token (<https://jwt.io>) 이용해서 Signature 처리를 수행해야 합니다.

This document clearly states that which API should be applied JWT in each API's description.

[VN]Tại tài liệu này, API có áp dụng JWT được giải thích rõ ràng tại phần mô tả API

[KO]이 문서에서는 JWT를 적용받는 API에 대해서는 API 설명 부분에서 명확하기 기술합니다.

## vani JWT Setting

In this section, JWT setting defined by vani is described.

[VN]Cài đặt JWT định nghĩa bởi vani được giải thích tại phần này.

[KO]이 영역에서는 vani에서 정의한 JWT 설정을 설명 합니다.

JWT consists of 3 parts, separated by a dot (.), and each of the 3 parts consists of Header, Payload, and Signature. The settings that vani defines in each item are as follows.

[VN]JWT được cấu thành từ 3 phần và được phân tách bằng dấu chấm (.) và 3 phần được cấu thành lần lượt là Header, Payload, Signature. Cài đặt được vani quy định tại mỗi hạng mục này như sau.

[KO]JWT는 3개 부분으로 이루어져 있으며 점(.)으로 구분되고, 3개의 부분은 각각 Header, Payload, Signature로 구성됩니다. 각각의 항목에서 vani가 규정한 설정은 다음과 같습니다.

### **Header**

Signature algorithm defined by vani is HS512 . Therefore, you need to set the Header value as shown below.

[VN]Signature algorithm được định nghĩa bởi vani là HS512 . Theo đó, phải cài đặt giá trị Header như sau.

[KO]vani에서 정의하는 Signature algorithm은 HS512 입니다. 따라서 아래와 같이 Header 값을 설정해야 합니다.

```
{
  "alg": "HS512",
  "typ": "JWT"
}
```

JAVASCRIPT

The key required for the signature is provided by vani.

[VN]vani cung cấp Key cần thiết để Signature.

[KO]Signature를 위해 필요한 Key는 vani가 제공합니다.

### **Payload**

Payload contains the contents of the Request Body. **vani additionally requires the values of iss , aud , iat , and exp among Registered claims defined by JWT as mandatory.**

[VN]Nội dung của Request Body đặt trong Payload. **vani yêu cầu thêm giá trị của iss , aud , iat và exp trong Registered claim được định nghĩa trong JWT như một trường bắt buộc.**

[KO]Payload 에는 Request Body의 내용이 들어갑니다. **vani는 추가로 JWT에서 정의한 Registered claim 중 iss , aud , iat , exp 값을 필수로 요구합니다.**

### **Signature**

The signature is generated by connecting the result of base64Url-encoded Header and Payload described above with a dot (.). Vain's Signature Algorithm is HS512 , and the private key required for signature is provided by vani.

[VN]Signature được tạo thành khi kết nối kết quả base64Url Encoded Header và Payload được định nghĩa ở trên với dấu chấm (.). Signature Algorithm của vani là HS512 và Private key cần thiết trong signature được cung cấp bởi vani.

[KO]위에서 설명한 Header와 Payload를 각각 base64Url Encoded 한 결과를 점(.)으로 연결한 값으로 Signature를 생성합니다.vain의 Signature Algorithm은 HS512 이며 서명에 필요한 key는 vani가 제공합니다.

## filling vani coin(point)

It fills coin(point).

[VN]Nạp xu (point).

[KO] 포인트(코인)을 충전합니다.

### Endpoint (JWT)

POST /points-filling

## JSON Structure

Request Body

Property	Type	Necessity	Description
requestToken	string	Yes	Token generated by JWT based on Header, Payload, Signature [VN]Token được tạo từ JWT dựa trên Header, Payload, Signature [KO]Header, Payload, Signature기 반 JWT로 만들어진 Token

Example

```
{
  "requestToken":
    "eeyJhbGciOiJIUzUxMiIsInR5cCI6IkpXVCJ9.eyJpc3MiOiI0O0CIsmImF1ZCI6InZhbmktcG9pbmRzLWZpbGxpbmciLCJpYXQiOiJlE2NjIyNTM0MDAsImV4cCI6MTY2MjI1NDQwMCwicmVxdWVzdElkIjoimTI4MTdkZTQtYjFmMC00YzU5LTk2MGMtMDBjYjcxMTNjMTczIiwiaWY3VzdG9tZXJJZCI6ImN1c3RvbWVyMTIzMzQ0IiwiaWY1vdW50T2ZwYW5pUG9pbmR1b2ZpbGwiOjE5MDAwfQ.6FSZEWELg4E9vbBC9adgIgKEmVmc-1CKaAsdUpnRd7tsL3W6_d5BlzhVL-BPF-8Mc6iMjIFiLMgIJBGkt8seXg"
}
```

JSON

Payload

Property	Type	Necessity	Description
iss	String	Yes	Token issuer. Apply the membershipId given by vani. [VN]Là nơi phát hành Token. Áp dụng membershipId được cấp bởi vani. [KO]Token 발급기관을 의미합니다. vani에서 발급한 membershipId를 적용합니다.
aud	String	Yes	Fixed value. vani-points-filling [VN] vani-points-filling là giá trị cố định [KO] vani-points-filling 로 고정값
iat	Number	Yes	The time at which the JWT was issued.This value should be the value of <u>Unix timestamp</u> ( <a href="https://www.unixtimestamp.com/">https://www.unixtimestamp.com/</a> ). [VN]Thời gian JWT được phát hành. Giá trị này phải là giá trị <u>Unix timestamp</u> ( <a href="https://www.unixtimestamp.com/">https://www.unixtimestamp.com/</a> ). [KO]JWT가 발급된 시간입니다. <u>Unix timestamp</u> ( <a href="https://www.unixtimestamp.com/">https://www.unixtimestamp.com/</a> )값 이어야 합니다.
exp	Number	Yes	The time at which the JWT was expired. This time should be <b>10 minutes later than iat</b> . This value should be the value of <u>Unix timestamp</u> ( <a href="https://www.unixtimestamp.com/">https://www.unixtimestamp.com/</a> ). [VN]Thời gian hết hạn JWT. Phải <b>chậm hơn 10 phút so với iat</b> . Giá trị này phải là giá trị <u>Unix timestamp</u> ( <a href="https://www.unixtimestamp.com/">https://www.unixtimestamp.com/</a> ). [KO]JWT 만료 시간입니다. <b>iat보다 10분</b> 늦어야 합니다. <u>Unix timestamp</u> ( <a href="https://www.unixtimestamp.com/">https://www.unixtimestamp.com/</a> )값 이어야 합니다.
requestId	string	Yes	Unique request id generated by merchant



customerId	string	Yes	A Unique identifier of a customer (generated by Membership System)
amountOfVaniPointTofill	number	Yes	The amount of vani point to fill. It must be positive integer [VN]Vani Xu được yêu cầu nạp. Phải là số nguyên dương. [KO]충전요청하는 vani coin. 반드시 양의 정수로 전달되어야 합니다.
eventType	String	Yes	The triggering event of a merchant for a points filling. Fixed Value. SWAP : event by the merchant point's swap. REWARD :event by giving to a user of a merchant as reward [VN]Đây là chương trình được đối tác yêu cầu nạp Vani Xu. Giá trị cố định. SWAP : Chương trình đổi điểm thành viên của đối tác REWARD : Chương trình cấp ưu đãi cho khách hàng phía đối tác [KO]포인트 충전을 요청하는 제휴사의 이벤트 입니다.고정값. SWAP : 제휴사의 포인트 SWAP으로 인한 Event REWARD :제휴사의 고객 혜택 제공을 위한 이벤트

### Example

JAVASCRIPT

```
{
  "iss": "48",
  "aud": "vani-points-filling",
  "iat": 1662253800,
  "exp": 1662254400,
  "requestId": "12817de4-b1f0-4c59-960c-00cb7113c173",
  "customerId" : "customer123344",
  "amountOfVaniPointTofill": 12000
}
```

### Response Body

Property	Type	Always Exists	Description
transactionId	String	Yes	Unique transaction id of the request generated by vani
transactionTime	string	Yes	Transaction time. format: YYYY-MM-DDThh:mm:ssTZD ( <a href="https://www.w3.org/TR/NOTE-datetime">ISO-8601</a> (https://www.w3.org/TR/NOTE-datetime) ) Example: 2017-07-17T13:42:40+01:00

## Cancellation of Filling Transaction

It requests cancellation of a previous point filling request.

[VN]Yêu cầu hủy nạp xu

[KO] 충전된 포인트를 취소 요청합니다.

If a customer has used part of the filled amount, vani may reject the cancellation request.

[VN]Nếu khách hàng đã sử dụng một phần số xu đã nạp trước đó, vani có thể từ chối yêu cầu hủy nạp xu

[KO]만약 고객이 충전된 금액을 일부 사용했다면, vani는 충전 취소 요청을 거절 할 수 있습니다.

### Endpoint

## POST /points-filling-cancellation

## JSON Structure

### Request Body

Property	Type	Necessity	Description
requestToken	string	Yes	request Token generated by JWT based on Header; Payload, Signature

### Example

```

{
  "requestToken":
    "eyJhbGciOiJIUzUxMiIsInR5cCI6IkpXVCJ9.eyJpc3MiOiI0OClzImF1ZCI6InZhbmktcG9pbmRzLWZpbGxpbmciLCJpYXQiOiJlYNTM0MDAsImV4cCI6MTY2MjI1NDQwMCwicmVxdWVzdElkIjoiaWMTIi4MTdkZTQtYjFmMCC0YzU5LTK2MGMtMDBjYjIzMTNjMTczIiwiaY3VzdG9tZXJJZCI6ImN1c3RvbWVtMTIzMzQ0IiwiaY2FuY2VsbGF0aW9uUG9pbmQ0Ij0jEYMDAwfQ.S1XfAU_z1FP_8bhtGz35XghadFKaad4hIHaXRXS_p8ihDg7dsuoqE21gqTyAzPVTqcCCcPn0sZicPr1KerI4Gw"
}

```

## Payload

Property	Type	Necessity	Description
iss	String	Yes	Token issuer. Apply the membershipId given by vani. [VN] Là nơi phát hành Token. Áp dụng membershipId được cấp bởi vani. [KO] Token 발급기관을 의미합니다. vani에서 발급한 membershipId를 적용합니다.
aud	String	Yes	Fixed value. <b>vani-points-filling</b> [VN] <b>vani-points-filling</b> là giá trị cố định [KO] <b>vani-points-filling</b> 로 고정 값
iat	Number	Yes	The time at which the JWT was issued. This value should be the value of <u>Unix timestamp</u> ( <a href="https://www.unixtimestamp.com/">https://www.unixtimestamp.com/</a> ). [VN] Thời gian JWT được phát hành. Giá trị này phải là giá trị <u>Unix timestamp</u> ( <a href="https://www.unixtimestamp.com/">https://www.unixtimestamp.com/</a> ). [KO] JWT가 발급된 시간입니다. <u>Unix timestamp</u> ( <a href="https://www.unixtimestamp.com/">https://www.unixtimestamp.com/</a> ) 값 이어야 합니다.
exp	Number	Yes	The time at which the JWT was expired. This time should be <b>10 minutes later than iat</b> . This value should be the value of <u>Unix timestamp</u> ( <a href="https://www.unixtimestamp.com/">https://www.unixtimestamp.com/</a> ). [VN] Thời gian hết hạn JWT. Phải <b>chậm hơn 10 phút so với iat</b> . Giá trị này phải là giá trị <u>Unix timestamp</u> ( <a href="https://www.unixtimestamp.com/">https://www.unixtimestamp.com/</a> ). [KO] JWT 만료 시간입니다. <b>iat보다 10분</b> 늦어야 합니다. <u>Unix timestamp</u> ( <a href="https://www.unixtimestamp.com/">https://www.unixtimestamp.com/</a> ) 값 이어야 합니다.
requestId	string	Yes	Unique request id generated by a Merchant

orgRequestId	string	Yes	The request id of the previous filling request [VN]Mã duy nhất yêu cầu nạp xu được Merchant gửi khi yêu cầu nạp xu [KO]충전 요청시 Merchant가 전달한 충전요청 고유번호
cancellationPoint	number	Yes	The cancellation amount of points. The amount of point is identical of the previous requested filling amount of points. [VN]Số điểm yêu cầu hủy. Gửi đi số điểm tương tự với số điểm đã yêu cầu nạp hoặc sử dụng. [KO]취소 요청하는 Point. 충전 또는 사용 요청했던 금액과 동일한 금액을 전달합니다.

JAVASCRIPT

```
{
  "iss": "48",
  "aud": "vani-points-filling",
  "iat": 1662253800,
  "exp": 1662254400,
  "requestId": "12817de4-b1f0-4c59-960c-00cb2313c173",
  "customerId": "customer123344",
  "cancellationPoint": 12000
}
```

### Response Body

Property	Type	Always Exists	Description
transactionId	String	Yes	Unique transaction id
transactionTime	string	Yes	Transaction time. format: YYYY-MM-DDThh:mm:ssTZD ( <a href="https://www.w3.org/TR/NOTE-datetime">ISO-8601</a> (https://www.w3.org/TR/NOTE-datetime) ) Example: 2017-07-17T13:42:40+01:00

## Transaction Information of Points Filling

It retrieves the result of a filling transaction requested by a Merchant.

[VN]Tra cứu kết quả về yêu cầu nạp xu được Merchant yêu cầu.

[KO]Merchant에서 요청한 포인트 충전요청에 대한 결과를 조회합니다.

### Endpoint

GET /points-filling-info/{requestId}

requestId is the unique request id of /points-filling or /filling-cancellation API sent by a Merchant.

[VN]requestId là mã được gửi khi Merchant yêu cầu API /points-filling' hoặc /points-filling-cancellation .

[KO]requestId는 Merchat에서 /points-filling 나 /points-filling-cancellation API 요청시 전달한 requestId 입니다.

### JSON Structure

#### Request Body

None

#### Response Body

Property	Type	Always Exists	Description
status	string	Yes	Status of the transaction related to the requestId SUCCESS : successful, FAILURE : failure, PROCESSING : under processing and not finished yet [VN]Trạng thái xử lý transaction ứng với requestId. SUCCESS : Thành công, FAILURE : Thất bại, PROCESSING : Đang tiến hành [KO]requestId에 해당하는 transaction의 처리상태. SUCCESS : 성공, FAILURE : 실패, PROCESSING : 진행 중
transactionId	string	No	Membership system's transactionId related to the request id.
transactionTime	string	No	Transaction time of the request was completed. format: YYYY-MM-DDThh:mm:ssTZD ( <a href="https://www.w3.org/TR/NOTE-datetime">ISO-8601</a> ) ( <a href="https://www.w3.org/TR/NOTE-datetime">https://www.w3.org/TR/NOTE-datetime</a> ) ) Example: 2017-07-17T13:42:40+01:00



If no matching data regarding the requestId, the HTTP Status Code should be passed as **404** .

[VN]Khi không có dữ liệu tương ứng với requestId, HTTP Status Code phải là 404.

[KO]requestId에 해당하는 데이터가 없는경우 HTTP Status Code는 **404** 로 전달되어야 합니다.

Version 1.1.4

Last updated 2023-06-23 10:03:25 +0700