

CoinBridge Back-Office Connector

Introduction

Table of Contents

COINBRIDGE BACK-OFFICE CONNECTION INTRODUCTION	2
INTRODUCTION	2
REQUIRED PERFORMANCE METRICS.....	2
<i>SLA</i>	2
<i>Response Time</i>	2
DEFINITIONS	3
MAIN PROCESSES	3
BACK-OFFICE CONNECTOR REQUIRED ENDPOINTS	4
<i>Data-Fetching Endpoints</i>	4
<i>Data-Reporting Endpoints</i>	5
<i>iOS – Apple Pay User Event</i>	8

CoinBridge Back-Office Connection

Introduction

Introduction

The CoinBridge Back-Office connector connects the customer's loyalty or CRM platform with the CoinBridge platform.

This connection is imperative to allow CoinBridge dynamically draw information from customer's platform whenever a user makes a purchase (drives a transaction) and for the CoinBridge platform to provide transaction details feedback, back to the customer's platform.

To obtain a comprehensive technical guide on the back-office connector, please consult our **backend technical documentation** ([can be found within the CoinBridge Wiki](#))

Required Performance Metrics

SLA

Service Level Agreements (SLAs) and response time are crucial components for this connection, and its performance and availability can greatly impact the user experience and overall success of the connection between CoinBridge and the customer's loyalty.

An SLA with a monthly uptime guarantees of 99.96% is required (up to 17.5 minutes of downtime per month).

Response Time

The required API response time for each endpoint group varies as follows:

- **Data-Fetching** Endpoints - Required API Response time - up to 400ms.
- **Data-Reporting** Endpoints - Required API Response time - up to 60s.

Definitions

- Customer: You.
- User: Your customers who are using your loyalty, gift card or other such mobile apps and platforms, and are members of the respective programs.
- Spending policy(ies): A rule or a set of rules for the Users. An integral part of the program they are enrolled in that permits or limits their spending in any way or form.
 - A spending policy is a set of rules that details all allowances and restrictions relevant to a specific user at the time of transaction. These rules are set and managed by the customer on the customer's own platform.
 - For example, such policies may include (but are not limited to):
 - Restrict or allow certain merchant IDs (MID)
 - Restrict or allow certain merchant categories (MCC)
 - Spending limits per transaction
 - Etc.
- User Balance: The current available spending balance and currency of a specific user at the time of transaction.

Main Processes

Connection to the CoinBridge Platform allows for the following main process:

- Verification of the user's service eligibility.
- Retrieve the user's current balance (or balances - when dealing with multiple products per user)
- Retrieve the user's spending policies.
- Send transaction feedback and data, back to the customer's loyalty or CRM platform.
 - Actual transaction result, including full transaction details.
 - Transaction status update - Additional reporting endpoint, to send updates on a transaction that was already reported.
 - Refund / Reversal transaction feedback

Back-Office Connector Required Endpoints

CoinBridge has classified their Endpoints into two distinct groups:

- **Data-fetching** endpoints are time critical and utilized for retrieving data from the customer's management platform.
- **Data-reporting** endpoints are utilized for transmitting transaction feedback and additional data back to the customer's management platform.

The below outlines the endpoints which CoinBridge requires to properly support payment via mobile devices.

Data-Fetching Endpoints

1. User Verification (End-User validity check)

This endpoint will be used during the user authentication flow. Its purpose is to validate whether the user exists and is eligible for this service.

The CoinBridge Back-Office Connection will be triggered to perform this call once an “*Authentication*” call was initiated by the mobile app.

Example:

- a. Request (In URL)
 - i. User ID
- b. Response
 - i. True / False

2. Retrieve Currently Available Balance (End-User Balance Fetch)

This endpoint will be used to retrieve the user's currently available balance, immediately prior to the transaction taking place.

The CoinBridge Back-Office Connection will be triggered to perform this call once a “*PrepareToPay*” call was initiated by the mobile app.

Upon successful retrieval of the user's balance information, CoinBridge SDK will enable the mobile device NFC component to make a payment.

If the returned balance is only valid until a specific timestamp, it can be returned as an optional parameter (Valid Until).

This parameter should only be returned if there is a real purpose for it since incoming transactions after this point will be rejected without a new preparation process.

Example:

- a. Request (In URL)
 - i. User ID
- b. Response
 - i. Balance Amount
 - ii. Balance Currency Code
 - iii. Valid Until (Optional)

3. Retrieve Spending Policy (Optional) - End-User Spending Policy Fetch

This endpoint will be used to retrieve spending policies and related user rulesets managed via the customer's management platform.

The CoinBridge Back-Office Connection will be triggered to perform this call once a "PrepareToPay" call was initiated by the mobile app.

Upon successful retrieval of the user's spending policy (in addition to the balance information from endpoint #2), CoinBridge SDK will enable the mobile device NFC component to make a payment.

Supported Rule Types:

- i. Include / Exclude Country
- ii. Include / Exclude MID (Merchant ID)
- iii. Include / Exclude MCC (Merchant Category Code)
- iv. Min Transaction Amount
- v. Max Transaction Amount

Data-Reporting Endpoints

1. Transmit Transaction Feedback

This endpoint will be used to transmit transaction feedback and additional data back to the customer's management platform.

The transaction result (approved or declined) will include full transaction details.

It is important to note that the transaction amount will always be a non-negative value, with a minimum value of zero.

The response can be an acknowledgement, or an event ID generated in your customer management platform - The id of this event to be tracked in the future.

Example:

a. Request

- i. User ID
- ii. Transaction ID
- iii. Transaction Amount
- iv. Transaction Currency Code
- v. Transaction Date and Time
- vi. Transaction Status (approved/declined)
- vii. MCC Code
- viii. Country Code
- ix. Merchant Name
- x. Merchant ID
- xi. Decline Reason Code (in case of a declined transaction)
- xii. Decline Reason Description (in case of a declined transaction)
- xiii. Transaction Source (E-Commerce/wallet/Physical Card/other)

b. Response

- i. Event ID (Optional)

2. Refund/Reversal

An endpoint similar to transaction feedback will be used to transmit **refund** transaction feedback and additional data to the management platform of the customer, but with some differences:

- The refund transaction will appear as an independent transaction, with a unique transaction ID
- The refund transaction won't be connected to any historical transaction. (That might exist as baseline for the refund action)
- Refund amount, similar to regular transaction, will always have a non-negative value, with a minimum value of zero, but in this case, you will have to act differently and update the user balance accordingly. ("Credit" balance instead of "Debit" balance)

- The status for a refund transaction will always be **approved**. (Due to that, the request won't include the "Transaction Status" parameter)

The response from your management platform can be an acknowledgement or an event ID generated in your customer management platform - The id of this event to be used for tracking in the future.

3. Transaction status update (Transaction Cancellation Report)

This endpoint will be used to transmit transaction cancellation, on an already-reported **Approved** transaction, due to a problem that may occur.

Retrieving this update means that the actual transaction was declined (the user saw it was declined, same as the merchant), while CoinBridge connector initially reported this transaction as approved - due to a card network error).

This update overrides the earlier transaction feedback (earlier transaction ID will be included), that means that the approved reported transaction was actually declined.

Please note that by getting this transaction update, you will need revert the transaction and update the user balance. (In case any changes to the balance were already made).

The response can be an acknowledgement, or an event ID generated in your customer management platform - The id of this event to be tracked in the future.

iOS – Apple Pay User Event

In accordance with Apple's requirements, it is the customer's obligation to dispatch various email/text notifications based on the activities (or inactivity) of end-users.

the purpose of this endpoint is to autonomously transmit events to the customer's loyalty platform, whenever a specific trigger related to Card Provision and Usage occurs.

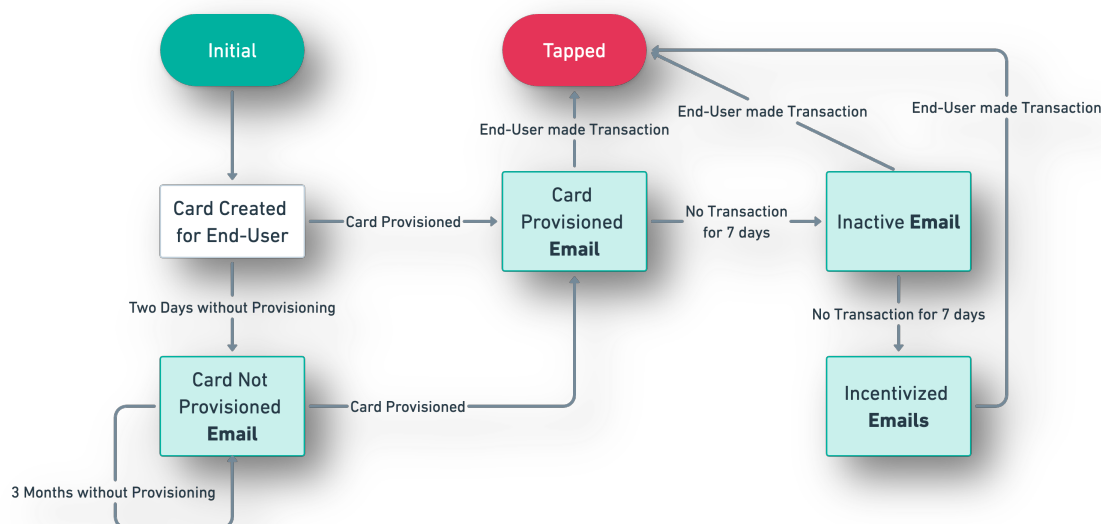
Card Provision denotes the act of adding the CoinBridge virtual card to the end-user's Apple Wallet on their device. (Relevant for the first successful provisioned device).

The Card Provision Status indicates whether the virtual card is currently present in the Apple Wallet or not.

Upon receiving an event from this endpoint, it is necessary for you to establish an automated process to send the appropriate emails to your end-users, adhering to the communication types outlined below.

Please be aware that sending these emails is a mandatory requirement.

Detailed information about the structure and content of these emails can be found in the CoinBridge **Solution Implementation Guidelines for iOS**.



additional information on the necessary communication types, please refer to the table below.

Event Type	Communication Trigger
CARD_PROVISIONED	User's card has been provisioned into an Apple wallet
CARD_NOT_PROVISIONED	User's card was issued but has not been provisioned into an Apple wallet for a predefined time (2 Days)
USER_INACTIVE	User's card was provisioned but no transactions has been made since for a predefined time (Within 7 days)
OFFER_INCENTIVE	A predefined monetary incentive should be offered to the user (Depending on the Customer's region, within 14 days)

Endpoint Structure:

- a. Request
 - i. Report ID
 - ii. User ID
 - iii. Event Type
 - iv. Event Metadata
 1. The Event Metadata will include the last four digits of the end-user's virtual card, and it is essential to include this information in the email sent to the end-user, as per Apple's requirements.
- b. Response
 - i. Event ID (Optional)