

# Card Management System

Quicko CMS is responsible for card issuing and account opening. The system can perform KYC and eKYC of the user before card issuing. You can manage card and user balance through CMS.

- [Introduction](#)
- [Overview](#)
- [Technical documentation](#)
  - [Mobile application](#)
  - [Server-server connection](#)
  - [Administration Panel](#)

# Introduction

To meet the needs and expectations of its customers, Quicko has developed a flexible infrastructure, allowing it to issue cards to fintech, merchants, companies, payment institutions or banks. We can provide digital issuing services for licensed payment and banking institutions or using our partner network, BIN sponsors to companies without payment licenses.

Quicko provides its customers with a range of services based on the applicable laws, directives and guidelines of card issuers such as Mastercard and Visa. Meeting these guidelines based on security standards including PCI DSS, 4 steps must be followed to deliver the card to the user:

## Create User

The first step to be fulfilled is to register the user in Verestro infrastructure in order to maintain his data in accordance with PCI DSS guidelines and secure the subsequent communication. Depending on the customer's needs to fulfill this step, there is a possibility of delivering a dedicated mobile application or, in case of already existing system, implementing SDK in own application or server-server connection in cases when the application is not necessary. Regardless of the path chosen, Verestro provides its customers with a dedicated Administration Panel to facilitate the management and monitoring of its customers.

## White Label Application

To meet the needs of the most demanding customers, Verestro has developed a mobile application for iOS and android. The application has a modular design which, in the shortest possible time, can be personalized to the required functionalities, branded according to the guidelines and published in production. More about this solution can be found in [White Label Application](#).

## Mobile SDKs

Customers with their own infrastructure and well-established products who want to provide their users with new mobile functionalities in a fast and easy way, including secure payment instruments, may use dedicated SDKs. Verestro team actively supports their implementation and leads through necessary certification processes. More on SDK based implementation can be found in [User Lifecycle & Card Management SDK](#).

## Life Cycle API

Customers who, similarly to the above case, want to expand their offer with competitive functionalities, where mobile application is not applicable, can use dedicated backend solution in server-server connection. LC API created for this purpose in a safe and easy to implement way allows to meet this requirement. More about LC API can be found in [User Lifecycle & Card Management API](#).

## KYC

In order to meet the requirements of card issuers, legal regulations and international directives Verestro supports the KYC (Known Your User) process aimed at verification of the customer to whom the services and payment instruments will be offered. As in the case of user registration, here, too, there is the flexibility of adjusting this solution both from the user's side in the mobile application and the processing of the application itself.

### Manual KYC Process

The standard Verestro solution makes it possible to collect the necessary data and photos of documents and persons in the mobile application and send the thus prepared request via a secure channel to the Verestro infrastructure. This process is supported both in the implementation of the [White Label Application](#) and/or implementation of SDK in the customer application. All KYC data are available through a dedicated Administration Panel, through which the client at a specific access level verifies the data submitted by users.

### Automatic eKYC

As KYC process requires customisations and flexibility, Verestro platform enables integration of external entities supporting this process. With the implementation of which it is possible to achieve full automation and thus reduce user verification time to a minimum. With KYC verification automation, the user can have a working payment device within 3-5 minutes of installing the application on their phone.

### External KYC

For institutions that are expanding their offerings to include card issuance and already have a KYC process in place, LC API is a dedicated channel from setting KYC status with the user more about it in [User Lifecycle & Card Management API](#).

## Create Balance

The third step that brings the user closer to obtaining the card is the creation of a balance / account for the user, which is a dedicated place that maintains the current balance of available funds in a specific currency. Depending on the customer's needs, the user can have a virtually unlimited number of balances.

## Automatic

The most commonly used solution is to automatically create the balance as soon as the user gets a positive KYC verification status. With this approach, the user receives the balance in the currency defined within the project.

## Manual

A client can create a balance for a user on demand or enable the user to do so themselves. Regardless of the implementation method, the process of creating balances is available in a dedicated mobile application, the provided SDK, from the server-to-server connection and through a dedicated admin panel. More on balance management can be found in [Card Management System](#).

## Create Card

The final step is to create a card linked to the previously created balance. Verestro provides its customers with the ability to generate virtual and physical cards for its users. With the implementation of the application in the minimum configuration specified by Mastercard, Verestro enables joining the Digital First program, where the user has a modern e-banking system along with a stylistically attractive physical representation of a virtual card. Processes related to issuing and managing cards are available in [Card Management System](#).

# Overview

Quicko Card Management System is called ANTACA. The platform provides solutions for creating and managing users' accounts (called "balances"), processing eKYC (user authentication process) and issuing payment cards generated for them.

CMS Antaca provides dedicated services for:

- end-user mobile applications,
- server-to-server connections helpful in integration with existing customer databases,
- administrative panel, necessary from the point of view of financial institutions in the process of issuing cards and managing their clients' funds.

CMS Antaca supports all necessary use cases for various digital and plastic card issuing. It supports integration with multiple issuing processors and can be connected with the one chosen by Verestro partner.

## Introduction to Card Issuing process

With the CMS Antaca you can offer your customers three types of cards:

1. Virtual card - Digital card without any physical components.
2. Physical card - The traditional plastic payment card.
3. Digital First card - Physical copy of your virtual card.

To be able to issue a card for a user, 4 requirements must be met:

1. You have to be integrated with Verestro platform using JWE token (described below) or other integration methods (API, SDK, White Label).
2. User must exist in Verestro database called DataCore. Make sure you register user via [User Lifecycle API & SDK](#).
3. User must be strongly verified according to KYC. You can use Verestro KYC (see below) or own KYC process.
4. The user must have a User Balance under which the card will be generated.

After those 4 steps you can issue a card for the User.

Below we describe this process step by step:

- Step 1. Configuration & JWE Security,
- Step 2. User Lifecycle API & SDK,
- Step 3. User registration & KYC,

- Step 4. Create User Balance (account),
- Step 5. Card issuing.

# Terminology

Name	Description
Customer	Institution which is using Quicko products. This institution decides which SDK should be used and how transaction should be processed. Basically Customer can be called Verestro client.
User	User which is using Payment Hub Application. It is root of entity tree. User is identified in Wallet Server by some unique identifier which is provided after registration. User can have access to his data and operations based on session. User's session is created after device pairing is performed. When session expires then user authentication have to be performed. Session is valid 10 minutes, however it is configurable parameter.
Card	Card belongs to the user. User can have many cards. Card is identified via internal id given after storing card on Wallet Server. Whole PAN is stored on Wallet Server which has PCI DSS certificate.
Device	Device belongs to user. When user starts using application after installation then device pairing is performed. After pairing device with some unique id, unique device installation id is generated and this installation is assigned to user. It is possible to have one active installation on specific device for specific user.
Session Token	Token which defines User. It is an authorization way of the User. This entity is created after pairing device and this is needed to perform any actions in the application. When session is expired then user authentication needs to be performed. Session is valid 10 minute s, however it is configurable parameter.
Sender	Quicko Wallet user which triggers transaction to the Receiver (check User description).
Receiver	Receiver can be identified in Wallet Server (Internal) or may be an entity that does not exist in Wallet Server (External) <ul style="list-style-type: none"> <li>◦ Internal - this type of Receiver has his own unique identifier just like sender. It can also act as a Sender in the transaction process,</li> <li>◦ External - this type of Receiver does not exist in Wallet Server. Transfers that are made to this type of Receiver require the entering of his card data by Sender</li> </ul>

Mid	Merchant identifier. This entity is representing Merchant in Acquirer's system. Customer have to provide the mid information to enable mid configuration in the Verestro system. Required to process 3DS authentication via Verestro System.
Acquirer	External institution responsible for processing transaction and 3ds requests ordered by the Verestro Payment Hub App. Acquirer connects with banks / card issuers and returns information whether the ordered action on a given card is possible.
PAN	It is 7-15 digits of credit card number. These digits contain the Permanent Account Number (PAN) assigned by the bank to uniquely identify the account holder.
Wallet Server	Provides the backend services to support Mobile Payment Application via Quicko Wallet SDK and is responsible for managing users, devices, cards , device tokens, storing transactions history and communication with Acquirers.
PCI DSS	PCI DSS (Payment Card Industry Data Security Standard) is a security standard used in environments where the data of payment cardholders is processed. The standard covers meticulous data processing control and protection of users against violations.
IBAN	IBAN (International Bank Account Number) is an international standard for bank account numbering that allows you to transfer funds to foreign accounts and to receive transfers from foreign entities to domestic bank accounts. One of the assumptions of the IBAN standard is to simplify the system of cross-border transfers.
QR	A QR code (quick response code) is a two-dimensional barcode.

---

## Configuration & JWE Security

To start the implementation, it is necessary to configure the payment processor. If we are using issuing processors already integrated with Quicko the process is simple and after quick information gathering (name of partner, BIN range, currency, remoteURL) a new card program can be setup for our partner.

The systems offered by Verestro are fully secure, which is confirmed by current third-party audits and PCI DSS certificates. As we store card and payment data we are obliged to comply with strict legal requirements. Card data are stored in a specially designed environment of Verestro called Data Core. This environment is placed in PCI Zone as well and is fully PCI DSS and GDPR certified.

You can communicate with the CMS Antaca API in three different dedicated channels:

1. Mobile Application - Methods starting with /Customers : designed for the mobile applications that use a session token sent in the header of each request. More about the possibilities of generating these tokens in the section White Label Application Overview.
2. Server-to-server - methods starting with /Secure : this communication channel is protected by the x509 certificate. To start an implementation based on this communication channel, it is necessary to generate your own CSR and send it to Verestro. Verestro will sign it and return a valid certification in a response.
3. Administrator and Customer Service (rarely used by partners) - methods starting with /admin : designed for the administration panel provided by Verestro.

## Additional data encryption & integration

Some requests and responses contain sensitive data, to additionally secure the connection we require JSON Web Encryption (JWE).

	normal	encrypted
Example of request with sensitive data	<pre>{   "cardNo" : "5555444455554444" }</pre>	<pre>{   "payload" : "very long JWE token" }</pre>
Example of response with sensitive data	<pre>{   "id" : 1125,   "type" : "1125",   "cvv" : "123",   "cardNo" : "5555444455554444",   "exp" : "2026-01-31" }</pre>	<pre>{   "payload" : "very long JWE token" }</pre>

## JWE configuration

To setup connection we need from you enc and alg from JWE parameters. Acceptable values are:

- Algorithm used by Verestro to encipher content of message (enc) - A256GCM,
- Algorithm used by Verestro to encipher encryption key (alg) - RSA-OAEP-256,
- Algorithm needed from you to encipher content of message (enc) - A256GCM,
- Allowed algorithms for key encryption (alg) - RSA-OAEP-256 or RSA-OAEP.

Recommended JWE libraries for various programming languages:

- [PHP](#),
- [JAVA](#).

Request

To process encrypted message you need to perform a few additional steps on top of standard message processing:

- Add headlines:
  - Public-Key through which you can transfer to us your public key encoded b64 (more details below),
  - Encrypted-Request headline confirming message encryption; value true or false,
- Download Verestro Public Key - see in technical API specs on which endpoint,
- Use Verestro Public Key to create JWE and transfer data table in payload,
- Use token (string) received in Verestro response in point 3 below key encryption key in payload.

Additional information:

- for GET methods avoid point 2, 3, 4 above (headlines mentioned in point 1 are still necessary),
- for empty POST methods (without "body") use same rules as for GET message.

Response

After sending to CMS Antaca encrypted request you will receive from us encrypted message:

1. Decipher token, which can be found in response below payload key (use your private key to perform this action),
2. After decipher action you can see response in unencrypted form.

Additional information:

- Response are encrypted only in case of success - HTTP 20X,
- The only exception from the above mentioned rule is code 204 No content,
- In case of errors (i.e. validation errors) you will receive unencrypted response,
  - ENCRYPTION\_REQUIRED,
  - INVALID\_PUBLIC\_KEY,
  - INVALID\_PAYLOAD,
  - CANT\_DECRYPT\_PAYLOAD.

Example request

<b>Correct request</b>	<b>Sent request (incorrect)</b>	<b>Received by CMS Antaca (after decipher action with private key)</b>
------------------------	---------------------------------	--

<div style="border: 1px solid black; width: 100%; height: 30px; margin-bottom: 10px;"></div> <div style="border: 1px solid black; width: 100%; height: 15px; margin-bottom: 10px;"></div> <p>1</p> <div style="border: 1px solid black; width: 100%; padding: 2px;">{"card_no" : 1337}</div>	<pre> {"payload" : "eyJhbGciOiJSU0EtT0FFUC0yNTYiLCJlbmMiOiJBMjU2R0NNIn0.rdUrW12XCZQgLFDJ-2zAHWYYnaAanctceE1-Y6yJUplX0B2dLu-bvYOEJ83KxxUs-ZjA41R4PmAVilx1cTF4pv-7CZR0_ki85XRATBYF2-MvZdcC81fHy2QPU_ZsAEWAW00a1wKJmuEsgPB2m1aLZ7oK4fC1hciep4PyAtuWQRYHjhNb-UDT41_gDKTbnSGTwhel7S0mAJ_HsKfnZFHUrM77UcxQGZKnH7Mzqvndf9THiMo0-3MWliYFDAm1bqN2_KTloBNCprYjFnylXPCjib73bjWX_P2ip5UI84cngbQmFVzc7o91JrpjvYou1INS7zL4XKLFcADN4nZ_9ePWsm5_kX5SOMyUyEhOC9gusrLNAJ0MHaIFHni8WqnMAWM3_MC4OQDYetKax5bnHK6x42_5eFaf6ZmzmioKny5aGm-4Vo8TEu691FmPxglhyenWIMhvBvf6ZeVsy58Ofr0mi3TXjwYbAyas7m6sncxZu1FhEj4da6gtNjmjuKdikOOntu8V71QQ07nczNqfGIUv0RcUc9uKjq5je4b9BEbK9WuQcroxmALqC4HTt1xhICHrVUA0d_t3fglhS2n7wNaKKCFq70ZWlrpdTaBd35kdVQOEjZgCavSjbZOzgOzcEqS6P2Blm7bZ7ZZBmnfk8y8M4m0xWoQNTmLC6nqz9bSbME.UERYKNCIDxQZpyWu.6Lw_5CcZ9HiVxHfi_XTAFw.pYbQ6tdmQYe1kiPonm1GhA"} </pre>	<div style="border: 1px solid black; width: 100%; height: 30px; margin-bottom: 10px;"></div> <div style="border: 1px solid black; width: 100%; height: 15px; margin-bottom: 10px;"></div> <p>1</p> <div style="border: 1px solid black; width: 100%; padding: 2px;">{"card_no" : 1337}</div>
--	--	--

Example response

<b>Correct response</b>	<b>Sent response by CMS Antaca</b>	<b>Received by you</b>
-------------------------	------------------------------------	------------------------

<div data-bbox="113 107 539 188" style="border: 1px solid black; height: 36px; width: 267px;"></div> <div data-bbox="108 241 544 264" style="border: 1px solid black; height: 10px; width: 273px;"></div> <p data-bbox="108 389 124 412">1</p> <div data-bbox="108 421 544 456" style="border: 1px solid black; padding: 2px;">{"card_no" : 1338}</div>	<pre data-bbox="576 107 1007 1406">{"payload" : "eyJhbGciOiJSU0EtT0FFUC0yNTYiLCJlbmMiOiBMbmU2R0NNIn0.iPmvEKtMAMrrEiR89vIwsl77ZfqXrcMiy- bx3z6_7HAo_aQzBpMVDtLyj3kTHYWxen8bhPuVyeBxYalHL20sekFzcIFFzvaGoyQYU6zOK8tPv81tgixQe8SDnEr5v9VWBfHxtPvqlpQlig2is5ynBkyqjdpQWEa gR3MpqpATGI7f- omG82Jq0OwZByWI8I6P89hczwgK37F -MUnQDxcRUM3RagbHKNeIcfmPdJpNeqFZHe45y4wUkTWN0uzW72qydkN_4uM9fy0nrUpgsJNbtJGAVIUvmDz4pIZki1 zyGbfZX-PT7Wh9UNM06gEUf4i2goZY- m4wPB0n2zXvxzcEdfTH27iPp- aKijfjPb_ZnHyklk_gZlAy9r7W0594d Y- eBJ_iUa5aeDsFS2TlfsfjMjsL8NRWY2noi Tw5IsneD8dwvr6N_rYcWoFXDyWXHo RitSSd2iYrB80gbeSOBw0wfKtPxNIZrR 0uDhke8FouS5Pk7QBw412kd43GtrEp Aijqn3ne7MNUpCtuNfj8e_NdGDLTR7C SHhC0jfFichplvklF42o216NO- Onyjsjdv1w4_w1ugs61fTHDI8IgbalOjO xauKwlvJJoyFdWmpjlXuzJhrray7ov25u h2ibvFv3Gfd2iuGUuLIZzYBOTT8ftGWT CGXTDvVOvzGbs.c3qMNb2Bne- 7g0Wz.PlngHFM6Q8Gn0p4Tlebig32s- ZrpLqTMqQDlpXLLYx0iq- StrKco_HrjdN4MxondP4CicCgseIjcV8JR 29jKYX- nqKdchEYq_vVlzFHcNI_Mx7y1eI192Qb Myx6b0Gbj5L79wpuB7qCUqTBNhJZ2c 07PuyPsewcNwglvnc-OrA- 2vL6lJnBi5ZGH8gBH1cZCgmbmMpZGN FPG3oFpOn9JPzmnvQxe9tvSFFj5989A 8d_XMHP- ZQ.djZxnBRxjeMKswDsCA3cXA"} </pre>	<p data-bbox="1045 107 1460 174">Check yourself by using Private Key included in the response.</p>
---	--	--

# User Lifecycle API & SDK

Once Verestro configured a project for your program and you are ready to authenticate with us using JWE token you will need to register users on our platform.

Please check the following components:

- If you want to integrate directly from mobile applications or integrate server-to-server - User Lifecycle and Card Management API & SDK.

# User registration & KYC

Once you registered users on our platform and would like to create accounts and issue cards for them you need to perform KYC. There are three alternative scenarios:

- You can use the Verestro KYC API in the verification process of your users.
  - Users can register from the level of the mobile application using the SDK method [/customers/me/register](#).
  - You can also use a dedicated method in the server-to-server connection to initiate the verification of your users [/secure/customers/\(customerid\)/register](#).
- If you already have KYC verification process on your side, just update the KYC flag for the user using User Lifecycle & Card Management API.

Once you registered users and performed KYC you can initiate account (called "balance") creation.

## Create User Balance

It is main account balance that is connected with user account and card. Main User Balance attributes are currency, balance value and balance state. In order to create User Balance make sure user got through KYC process. KYC process can be either manual or automated. It can be performed by partner or Verestro. It is highly recommended that User Balance is hold by Verestro but we can approve projects where partner holds User Balance.

In order to create any payment card at Verestro CMS you have to create User Balance first. Payment card issued for particular User Balance cannot be moved to another balance later.

There is an important rule - one user can have multiply balances and for every balance user can have multiply payment cards.

To create User Balance use the following methods:

- in case of server-to-server connection [/secure/customers/\(customerid\)/balances](#),
- in case of integration through mobile application [/customers/me/balances](#).

For more information about account / balance management please check technical APIs.

## Card issuing

With the Antaca API you can offer your customers three types of cards:

- Virtual card - Digital card without any physical components.
- Physical card - The traditional plastic payment card.
- Digital First card - Physical copy of your virtual card.

To be able to issue a card for a user, 3 requirements must be met:

- User must exist in a PCI DSS compliant **Data Core** system in Verestro. Make sure you register user via [User Lifecycle API & SDK](#). - **link**
- User must be strongly verified according to **KYC**. You can use Verestro KYC or own KYC process.
- The user must have a **User Balance** under which the card will be generated.
- After those 3 steps you can issue a card for the user.

## Digital card

If the API receives the request, it will create a 16-digit PAN (Permanent Account Number), CVC2 (Card Verification Code), and Expiry Date. You can then deliver this information to your customer.

## Physical card



## Digital-first card

CMS Antaca has been designed to meet the requirements of the Digital First program belonging to Mastercard, which contains guidelines for payment and e-banking applications to maintain the highest level of security, convenience of use in terms of accessibility and payment solutions from the level of the telephone and other devices.

The digital-first card is a physical copy of a virtual card that reflects its actual condition. It can be produced as a blank plastic with no data imprinted on it.

## Actions

### Create virtual

This method enables creation of virtual payment card for already created user and balance.

Availability

Collection	URL	Authentication	Encryption required	Available for admin roles
Customer	<b>POST</b> <a href="#">/customers/me/cards/virtual</a> or for an asynchronous process <b>POST</b> <a href="#">/customers/me/cards/virtual/async</a>	Session token	YES* - JWE  *for an asynchronous process NO	N/A
Admin	<b>POST</b> <a href="#">/admin/customers/{customerId}/cards/virtual</a>	Session token	YES* - JWE	Admin, Manager
API	<b>POST</b> <a href="#">/secure/customers/{customerId}/cards/virtual</a> or for an asynchronous process <b>POST</b> <a href="#">/secure/customers/{customerId}/cards/virtual/async</a>	x509 certificate	YES* - JWE  *for an asynchronous process NO	N/A

## Lock

This functionality enables temporary or fixed blocking of already issued cards. After card being blocked every authorisation request will be rejected. While using this method you need to inform CMS Antaca about reasons of card blocking. List of reasons is described below in the table.

Code No	Card stop reason	Failure Action code on POS/ATM	irreversible
1	Card lost	2008	YES
2	Card stolen	2009	YES
3	Pending query	1000	NO
4	Card consolidation	1016	NO
5	Card inactive	1018	YES
6	PIN tries exceeded	1006	NO
7	Suspected fraud	1002	NO
8	Card replaced	1011	YES

## Availability

Collection	URL	Authentication	Encryption required	Available for admin roles
Customer	<b>POST</b> <a href="#">/customers/me/cards/{cardId}/lock</a>	Session token	NO	N/A
Admin	Administrator blocks the card through the core of the administration panel	N/A	N/A	N/A
API	<b>POST</b> <a href="#">/secure/customers/{userId}/cards/{cardId}/lock</a>	x509 certificate	NO	N/A

## Unlock

This functionality enables unblocking previously blocked card. It works in case the card was not blocked with Code No 1, 2, 5 or 8 described in the above table (card lost, card stolen, card inactive or card replaced). CMS Antaca does not need reasons for card unblocking.

## Availability

Collection	URL	Authentication	Encryption required	Available for admin roles
Customer	<b>POST</b> <a href="#">/customers/me/cards/{cardId}/unlock</a>	Session token	NO	N/A
Admin	Administrator unblocks the card through the core of the administration panel	N/A	N/A	N/A
API	<b>POST</b> <a href="#">/secure/customers/{userId}/cards/{cardId}/unlock</a>	x509 certificate	NO	N/A

## Remove

This functionality enables card deletion from CMS Antaca. Deleted card cannot be restored.

## Availability

Collection	URL	Authentication	Encryption required	Available for admin roles
Customer	<b>DELETE</b> <a href="#">/customers/me/cards/{cardId}</a>	Session token	NO	N/A
Admin	Administrator delete the card through the core of the administration panel	N/A	N/A	N/A
API	Other APIs remove the card via LC or directly in DC	N/A	N/A	N/A

## Get full data

This functionality enables receiving full card data (PAN, Expiry Date, CVC2 or CVV). Access to those data for user should be always connected with additional authorisation by user (fingerprint, application PIN).

### Availability

Collection	URL	Authentication	Encryption required	Available for admin roles
Customer	<b>GET</b> <a href="#">/customers/me/cards/{id}</a>	Session token	YES	N/A
Admin	Administrator cannot view the full details of the cards	N/A	N/A	N/A
API	<b>GET</b> <a href="#">/secure/customers/{customerId}/cards/{id}</a>	x509 certificate	YES	N/A

## Reset CVV

This functionality enables generation of new CVC2 or CVV number for virtual or digital first cards.

### Availability

Collection	URL	Authentication	Encryption required	Available for admin roles
Customer	<b>POST</b> <a href="#">/customers/me/cards/{cardId}/cvv</a>	Session token	YES	N/A

Admin	<b>POST</b> <a href="#">/admin/cards/{cardId}/cvv</a>	Session token	YES	Admin, Manager, Employee
API	N/A	N/A	N/A	N/A

## Order physical card

This functionality enables ordering plastic card. Process of card personalisation can take up to 48 hours depending on chosen personalisation center. Additionally card will be transferred to user by courier or post office. Physical card ordered by this functionality will be inactive until activation action.

### Availability

Collection	URL	Authentication	Encryption required	Available for admin roles
Customer		Session token	YES	N/A
Admin		Session token	YES	Admin, Manager, Employee
API		x509 certificate	YES	N/A

## Link card

Around 48 hours after card ordering it will be visible in user resources. After Verestro receives confirmation from personalisation center that card was personalised CMS Antaca connects card with user account and balance. From this moment it can be visible for user and can be activated.

## Print virtual

Functionality available only for programs registered in Digital First by Mastercard. It enables creation of physical copy of already existing virtual card. Digital First card does not contain card data or user data on its plastic. It does not require activation. For more information please contact us.

### Availability

Collection	URL	Authentication	Encryption required	Available for admin roles
Customer		Session token	YES	N/A
Admin		Session token	YES	Admin, Manager, Employee
API		x509 certificate	YES	N/A

## Set PIN

This functionality is available for physical and Digital First cards. It enables setting up PIN that is used for face-to-face transactions (POS and ATM).

**IMPORTANT:**

- After setting up new PIN it is required to perform standard chip & pin transactions (recommended on ATM) to transfer PIN to chip on the plastic to be able to process off-line PIN transactions.
- Majority of POS terminals verifies offline PIN what can result in message "Incorrect PIN" on terminal. User should be informed about it.
- In case of contactless transactions online PIN will be used in all cases so user will not receive "Incorrect PIN" message on terminal.

Availability

Collection	URL	Authentication	Encryption required	Available for admin roles
Customer	<b>POST</b> https://prepaidapi.upaid.pl/customers/me/cards/{cardId}/pin	Session token	YES	N/A
Admin	N/A	N/A	N/A	N/A
API	N/A	N/A	N/A	N/A

## Activate card

This functionality enables activation of previously ordered physical card. Card transactions will not work until card is activated.

Availability

Collection	URL	Authentication	Encryption required	Available for admin roles
Customer		Session token	NO	N/A
Admin		Session token	NO	Admin, Manager, Employee
API		x509 certificate	NO	N/A

## Lock outside

This functionality enables blocking of card in CMS Antaca on request of external entities (MC or VISA or acquirers). It can be used in case user entered incorrect PIN 3 times or in other fraud related actions. This lock cannot be removed if card was blocked by Code No 1, 2, 5, 8 (see below). The table below contains all possible reasons of card lock.

Code No	Card lock reason	Failure Action Code on POS/ATM	Irreversible
1	Card lost	2008	YES
2	Card stolen	2009	YES
3	Pending query	1000	NO
4	Card consolidation	1016	NO
5	Card inactive	1018	YES
6	PIN tries exceeded	1006	NO
7	Suspected fraud	1002	NO
8	Card replaced	1011	YES

# More information on Partner Balances and Deposit Requirements

## Partner Balance

Partner Balance is used in the projects with Verestro and enables additional security of transaction funding for Verestro.. Main Company Balance attributes are currency, balance value and balance state.

Company Balance is used for transaction settlements in case of foreign exchange transactions especially in situations where there are differences between authorisation and clearing amounts. Verestro will setup Partner Balance for every project.

## Partner Credit Balance

Partner Credit Balance is used to process transactions of Partner especially in cases where User Balance is hold by Verestro. Examples of such projects are many standard projects where Partner is not financial institution or e-Wallet and does not hold User Balances on its side.

The main reason to use Partner Credit Balance is limiting transactions performed by Partner's users to funds hold on Partner Credit Balance. Verestro and its BIN sponsors cannot risk processing transactions without having funds available so this deposit needs to be used to enable transactions in such cases.

Partner through Verestro Administration Panel has access to actual level of Partner Credit Balance and can reload it by sending banking transfer to BIN Sponsor cooperating with Verestro. Partner can receive notification via e-email if Partner Credit Balance goes below pre-defined level.



## Partner Deposit Balance

Partner Deposit Balance is used alternatively to Partner Credit Balance. Partner Credit Balance is used to process transactions of Partner especially in cases where User Balance is not hold by Verestro. Examples of such projects are the ones with other wallet providers that already hold user balance or project where Verestro through its partners acts as BIN Sponsor or Principal Member for Affiliate Partner.

The main reason to use Partner Deposit Balance is limiting transactions performed by Partner's users to funds hold on Partner Deposit Balance. Verestro and its BIN sponsors cannot risk processing transactions without having funds available so this deposit needs to be used to enable transactions in such cases.

Partner through Verestro Administration Panel has access to actual level of Partner Deposit Balance and can reload it by sending banking transfer to BIN Sponsor cooperating with Verestro. Partner can receive notification via e-mail if Partner Deposit Balance goes below pre-defined level.



## Balance Summary in Administration Panel

Summary Balances are a control tool used for accounting and liquidity verification reasons. They are presented in Administration Panel in every currency used in the project.

### Users

Presents sum of all User Balances in particular currency.

### Wallet

Presents sum of all User Balances and all Partner Balances in particular currency.

## Actions

### Create user balance

This functionality enables creation of user balance in particular currency.

## Create Partner Deposit Balance or Partner Credit Balance

Not used in standard projects. This functional enables Partner creation of new Partner Deposit Balance or Partner Credit Balance for particular projects.

## Get User Balance

Enables getting user balance and list of cards connected to this balance (account).

## Get Partner Deposit Balance or Partner Credit Balance

This functional enables Partner getting information of Partner Deposit Balance or Partner Credit Balance for particular projects.

## Reload Partner Deposit Balance or Partner Credit Balance

Not used in standard project. This functional enables Verestro to reload Partner Deposit Balance or Partner Credit Balance for particular projects. It is used by Verestro.

## Reload user balance

This functionality enables reloading User Balance.

# Other functionalities

You can find additional methods in API descriptions:

- [API used for server-to-server connections,](#)
- [API used for mobile application-to-server connections,](#)
- [API used for Administration Panel access \(rarely used by partners\).](#)

In case of questions please let us know.

# Technical documentation

Technical documentation

# Mobile application

@swagger=<https://cardissuing.upaidtest.pl/api/documentation/customer/api-docs-customer.json>

Technical documentation

# Server-server connection

@swagger=<https://cardissuing.upaidtest.pl/api/documentation/secure/api-docs-secure.json>

Technical documentation

# Administration Panel

@swagger=<https://cardissuing.upaidtest.pl/api/documentation/admin/api-docs-admin.json>