Digital Payments
for a Trusted World

WORLDLINE

iDEAL 2.0 Getting Started Guide

Status	Published
Author:	Worldline
Document date:	28 April 2023
Classification:	Confidential
Version:	1.3

Version history

Version no.	Version date	Status	Edited by	Most important edit(s)
1.0	18-07-2022	Published	Paula Pellegrini	Initial Version
1.1	19-12-2022	Published	Bhumika Dave	Minor Edits to Section 2.2.3 and Section 2.3.2
1.2	13-03-2023	Published	Bhumika Dave	Added Chapter on Production
1.3	28-04-2023	Published	Bhumika Dave	Added example under Test Steps Amended Section 2.3.3 and 2.3.4 on Notification Tokens.

Table of contents

1 General Information on the Test Environment

1.1 What are the Open Banking APIs?

The Open Banking API are fully fledged RESTful APIs enabling the initiation of iDEAL 2.0 payments. Developer tools like Postman or SOAP UI can be used to test the Open Banking API in a test environment. More information on the APIs can be found in the Open Banking Implementation Guide for iDEAL.

1.2 What should you start with?

We recommend starting with discovering the overall iDEAL 2.0-related functionalities with the Open Banking APIs by reading the Open Banking Implementation Guide for iDEAL and trying out the different functionalities and APIs in the test environment. The test environment is set up and configured with a series of test data to facilitate an easy and fast integration.

1.3 What functionality is available in the test environment?

The test and production environments are functionally identical. Our test environment enables you to test the integration with our Open Banking APIs for iDEAL 2.0. The test environment contains a simulator for the iDEAL Hub and provides simulated API responses. All data in the test environment is mocked and is to be used only for integration testing purposes.

The test environment for the Open Banking APIs supports the following iDEAL 2.0-related functionalities:

- Initiating payments:
    - Standard/regular iDEAL 2.0 payments without user recognition
    - Standard/regular iDEAL 2.0 payments with user recognition via debtor tokens
    - iDEAL 2.0 Fast Checkout payments
- Retrieving debtor preferences (for payment with user recognition via debtor tokens)
- Retrieving status (and other) information about payments (multiple payment statuses can be tested by using the correct test data - See Test Data section)
- Receiving payment notifications

Merchant/PSP authentication is a mandatory pre-step for any of the above functionalities and can be performed via the dedicated token retrieval API.

The test environment for the Open Banking APIs does not support the following functionality:

- User/consumer interaction pages for issuer selection and profile registration

1.4 Who can use the test environment?

Any merchant/PSP, with an active contract for iDEAL (with the relevant Acquirer), can use the test environment to integrate with the Open Banking APIs.

2 Test Environment Setup

2.1 Test environment access information

- The test environment can be reached under following base domain: https://routingservice-rabo.awltest.de
  - The full endpoints for all APIs can be derived from the technical documentation (swagger files)
- The Open Banking APIs rely on the authentication of the Merchant/PSP accessing the iDEAL 2.0-related APIs through an access token
  - The retrieval of an access token requires the integrating party to identify the Acquirer + Merchant/PSP/Sub-Merchant combination via the "Client" and "Id" headers in the POST/token API respectively
    - The Acquirer identification to be used in the "Client" header is the following: **RaboiDEAL**
    - The Merchant/PSP/Sub-Merchant to be used in the "Id" header can be retrieved in the Test Data Section.

Acquirer	Base Domain	Client (for token request)
Rabobank	https://routingservice-rabo.awltest.de	RaboiDEAL

2.2 Test Data

In order to test the connectivity to the Open Banking APIs in the test environment, the usage of pre-defined test data is necessary:

- Merchant/PSP test certificates
- Merchant/PSP/Sub-Merchant IDs corresponding to the test merchants available on the test environment
- Specific payment amounts to simulate different payment statuses

2.2.1 The certificates and the private keys

We have provided an example certificate that you can use to integrate with the Open Banking APIs in the test environment. These certificates will give you the possibility to retrieve an access token as a mandatory pre-step to consuming the payment APIs.

The TestCertificatesiDEAL2.0 file contains both the private and public keys needed in order to perform the integration tests.

2.2.2 The test merchants/PSPs

Test merchants have been set up on the test environment, in order to allow testing the integration with the Open Banking APIs depending on the underlying productive Merchant/PSP setup.

Following Merchant/PSP setups are possible:

- Merchant or cPSP with Sub-Merchants integrating directly or via a Service Provider (dPSP)
- Merchant with no Sub-Merchants integrating directly or via a Service Provider (dPSP)

Following test merchants are available on the test environment:

- **Test Merchant A - 002881**
  - Merchant setup with no sub-merchants
- **Test Merchant B - 002882**
  - Merchant/cPSP setup with sub-merchants
  - Sub-merchant Ids: 1, 2, 3, 4

The different Merchant/PSP setups and the corresponding test merchant to use in the test environment:

Merchant/PSP setup	Way of Integration	Test Merchant	Test Sub-Merchant Ids (one of)
Collecting PSP with Sub-Merchants	Direct Integration	B	1, 2, 3, 4
Merchant with Sub-Merchants	Direct Integration	B	1, 2, 3, 4
Merchant with no Sub-Merchants	Direct Integration	A	N/A
Collecting PSP with Sub-Merchants	Via dPSP*	B	1, 2, 3, 4
Merchant with Sub-Merchants	Via dPSP*	B	1, 2, 3, 4
Merchant with no Sub-Merchants	Via dPSP*	A	N/A

*dPSPs should know the underlying setup of the Merchant/PSP they are providing the connection for and use the appropriate test merchant/merchants (dPSPs may need to test both).

2.2.3 The payment amounts

Different payment statuses can be simulated in the test environment by providing specific payment amounts during the payment initiation (in the POST payments API):

Payment amount in POST /payments	Payment status
1,00	To retrieve a payment with the status “SettlementCompleted”
2,00	To retrieve a payment with the status “Cancelled”
3,00	To retrieve a payment with the status “Expired”
4,00	To retrieve a payment with the status “Open”
5,00	To retrieve a payment with the status “Error”

2.3 Test steps for initiating an iDEAL 2.0 payment

This section describes how to connect an application to Open Banking API for iDEAL in the test environment and execute iDEAL 2.0 payments.

2.3.1 Step 1 - Request an access token

When requesting an access token, you must authenticate yourself as a Merchant/PSP by providing your Merchant/PSP/Sub-Merchant Id and using your certificate. In the response, an access token is returned, with a validity of 60 minutes.

A Merchant/cPSP can also request for an access token on behalf of all SubMerchants which can then be used by each SubMerchant, provided they send in the InitiatingPartySubID field under Creditor Information in the `POST /payments` API.

In the test environment, both the Merchant/PSP/Sub-Merchant Id and the certificate to use are provided (see Test Data section).

Note: In the productive environment, certificates generated by the Merchant/PSP and productive Merchant/PSP/Sub-Merchant Ids are used.

2.3.1.1 Request attributes

Attribute/Header	Value for test environment
Client	*See under Test environment access information*
Id	*See under Test Data - The test merchants/PSPs*
App	IDEAL

2.3.1.2 Examples

For cPSP/Merchant with Sub-Merchants (Test Merchant B):

Attribute/Header	Value for test environment
Client	RaboiDEAL
Id	002882:1
App	IDEAL

The generated token here will be valid for the Sub Merchant (1) Specified under Id.

For cPSP/Merchant with Sub-Merchants (Test Merchant B):

Attribute/Header	Value for test environment
Client	RaboiDEAL
Id	002882
App	IDEAL

The generated token here will be valid all SubMerchants (1,2,3 and 4) under the Merchant/cPSP, provided they send in send in the InitiatingPartySubID field under Creditor Information in the `POST /payments` API.

For Merchant without Sub-Merchants (Test Merchant A):

Attribute/Header	Value for test environment
Client	RaboiDEAL
Id	002881
App	IDEAL

Full request-response examples and more information on the authorization `POST /token` request can be found in the Open Banking Implementation Guide for iDEAL (Chapter OB v3 for iDEAL - Authorization API).

Information on the digital signature required for the token request can be found in Open Banking Implementation Guide for iDEAL (OB v3 for iDEAL - Security).

2.3.2 Step 2 - Initiate an IDEAL 2.0 payment

Use the access token retrieved in Step 1 to initiate an iDEAL 2.0 payment.

The necessary fields to use in the POST /payments API depend on the functionality desired:

- Regular/standard iDEAL 2.0 payments with no profile recognition - available in test environment
- Regular/standard iDEAL 2.0 payments with profile recognition via Debtor Tokens - will be available soon in test environment
- iDEAL 2.0 Fast Checkout payments – available in test environment

Full request-response examples and information on the applicable fields for the different functionalities can be found in the Open Banking Implementation Guide for iDEAL.

Initiating a payment at this stage will lead to a dummy Redirect URL of https://worldline.com. This is NOT a functional URL. To check the success of a test transaction, testing should be done with the payment notifications based on the payment amounts listed in Section 2.3.3. e.g. sending an amount of 1,00 € to get back “SettlementCompleted”.

Note: The next step will be receiving a payment notification with updates about the initiated payment. The pre-requisite for this notification is for the Status Notification URL to be provided via the MSP GUI Subscription along with the Notification Bearer Token. The URL can be overridden with a notification URL provided by the integrating party in the `POST /payments` in the "InitiatingPartyNotificationUrl" header field.

2.3.3 Step 3: Receive payment notification

Simulated notifications will be sent out in the test environment to Merchants/PSPs. As mentioned above, the Status Notification URL and Notification BearerToken are conditionally mandatory in the iDEAL MSP Dashboard for Merchants expecting to receive a notification status.

The Notification BearerToken is a Static token of choice set by the Merchant to validate the notifications sent by the TPP solution. In the case of the test Merchants, a dummy notification URL and Token are preset with the following values :

Field	Value
Notification URL	https://merchantdomain.com/notification/v3
Notification Bearer Token	iDEAL2.0testnotificationtoken

As the Notification URL in the iDEAL MSP Dashboard is a dummy value, the notification endpoint of the Merchant/PSP, where the notifications will be sent to, should be provided in the `POST /payments` API.

Please ensure this URL ends with /v3 as specified in the Open Banking Payment Initiation Service Swagger File. When this endpoint is provided, it must be implemented by the Merchant/PSP according to the specifications described in the `POST /status` API (see Technical documentation/Swagger files for exact specification).

Full request-response examples and information on the applicable fields for the different functionalities can be found in the Open Banking Implementation Guide for iDEAL.

2.3.4 Step 4 (optional): Retrieve payment status

Optionally, a Merchant/PSP can also retrieve the information about a payment by using the `GET payments/{paymentId}/status` API (see Technical documentation/Swagger files for exact specification).

Full request-response examples and information on the applicable fields for the different functionalities can be found in the Open Banking Implementation Guide for iDEAL.

3 Moving to Production

3.1 Productive Environment Setup

3.1.1 Changes to be made on iDEAL MSP Dashboard

Once you have successfully completed testing in the test environment as per the above mentioned steps, you are ready to move to Production. Before doing so, please check with your Acquirer on the additional information you may need to include in your iDEAL MSP Dashboard.

The following new fields would be visible to you under your iDEAL Subscription. Here is a short overview on what each of these fields refers to :

Field	Definition
cPSP Flag	This field indicates whether a Merchant is a cPSP provider.
Debtortoken callback URL	URL where the Debtortoken callback is received. This field can also be included in the `POST /payments` API call. (Part of Release 3). The value in API call will override that in the iDEAL MSP Dashboard. This is not ready for production.
QR code callback URL	URL where QR code callback is received. This field can also be included in the `POST /payments` API call (Part of Release 4). The value in API call will override that in the iDEAL MSP Dashboard. This is not ready for production.
Return URL	This refers to the Merchant Return URL. This field can also be included in the `POST /payments` API call. This is a mandatory field to have in either one of the locations. You may also send it in both. The value provided in the API will override the value from the iDEAL MSP Dashboard.
Status Notification URL	Notifications will be sent to the following URL. This field is required in the iDEAL MSP Dashboard if a notification status is expected by the Merchant. The URL can be overridden by including it in the `POST /payments` API
Notification BearerToken	Static token of choice set by the Merchant to validate the notifications sent by the TPP solution. This is conditionally mandatory if a status notification is expected by the Merchant.
CPSP Scheme ID	To be filled if a Merchant is a cPSP.
C2C Provider Scheme ID	To be filled if a (sub)Merchant is a C2C provider.
Country Code	Two-letter country code according to ISO 3166-1 alpha-2 standard. It indicates the origin country of the (sub)Merchant.
Merchant Category code (MCC)	The Merchant Category Code of the (sub)Merchant being paid for this transaction, following the ISO 18245 standard.

Important Info : Depending on your Acquirer, certain fields may be mandatory/optional.
Please check with your Acquirer for further information.

3.1.2 Productive Transactions

Once the iDEAL MSP Dashboard is enriched with the required information for iDEAL 2.0, you may proceed to make payments on the productive environment using the same steps as taken in Test Environment.
The certificate uploaded in the MSP iDEAL Dashboard will be used for the getting an access token via the Authorization API request. Please refer to the iDEAL 2.0 Implementation guide for more details.

To make transactions in the production environment, replace the base domain with that mentioned in the table below :

Acquirer	Base Domain	Client (for token request)
Rabobank	https://ideal.rabobank.nl	RaboiDEAL

The Initiating Party ID used in the Authorization API request will be the **Platform Merchant ID** from the MSP Dashboard. This is visible on the Merchant Screen, referred to as "Platform Merchant ID, Merchant ID or Dashboard ID" depending on your Acquirer. Please take note that the Service specific Merchant ID, as found under the iDEAL MSP Dashboard is **not to be used** for the Authorization Request. Doing so will lead to an error being generated.