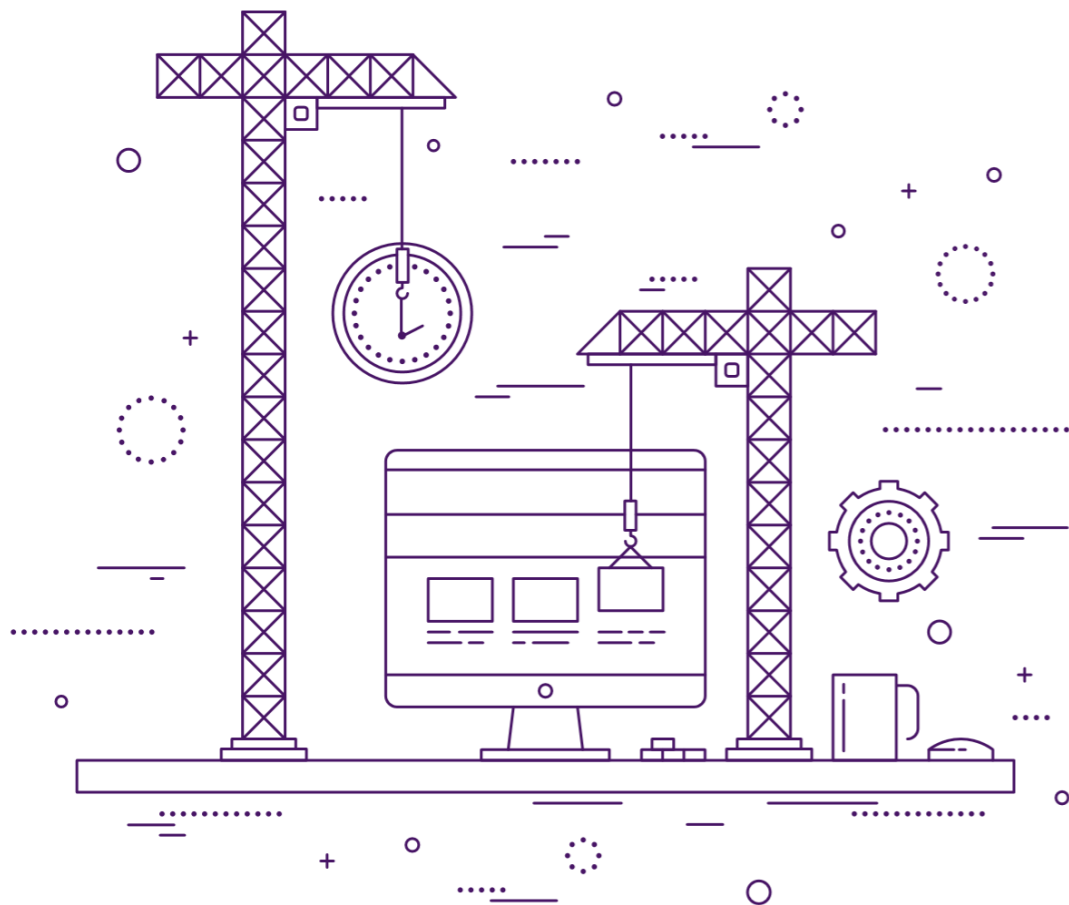# Integration Guide
for E-Commerce Platforms

v2.10, 10.03.2022

In this guide:

Introduction

Here at Inbank we strive to help our partners sell more by simplifying purchases and making financing more accessible to customers. For exactly this reason we offer a number of buy-now-pay-later (BNPL) solutions. Our most known credit offering is hire-purchase, also known as payment by installments.

There are several methods of how partners can integrate with Inbank, this document covers our **e-POS solution**. With Inbank e-POS, partners only need to add Inbank as a payment method and redirect clients to our environment, Inbank will take care of all the rest. After a successful financing process we will redirect the customer back to you.
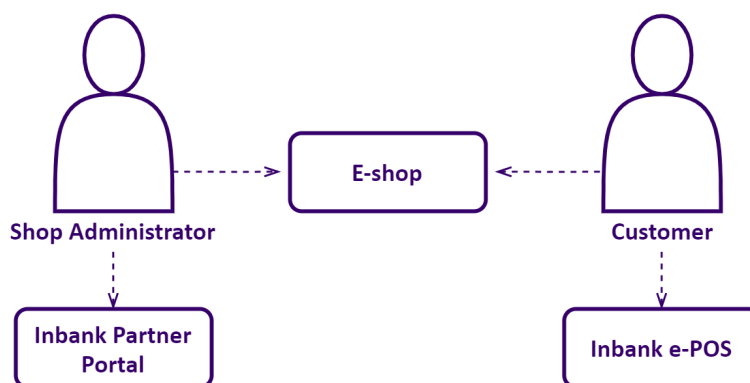
Inbank e-POS is supplemented with Inbank Partner Portal where merchants can see detailed overview of submitted credit applications, create applications for customers and conduct contract withdrawals.

For any questions regarding the e-POS integration process, contact Inbank at integration@inbank.ee. We will be happy to help.

Changelog

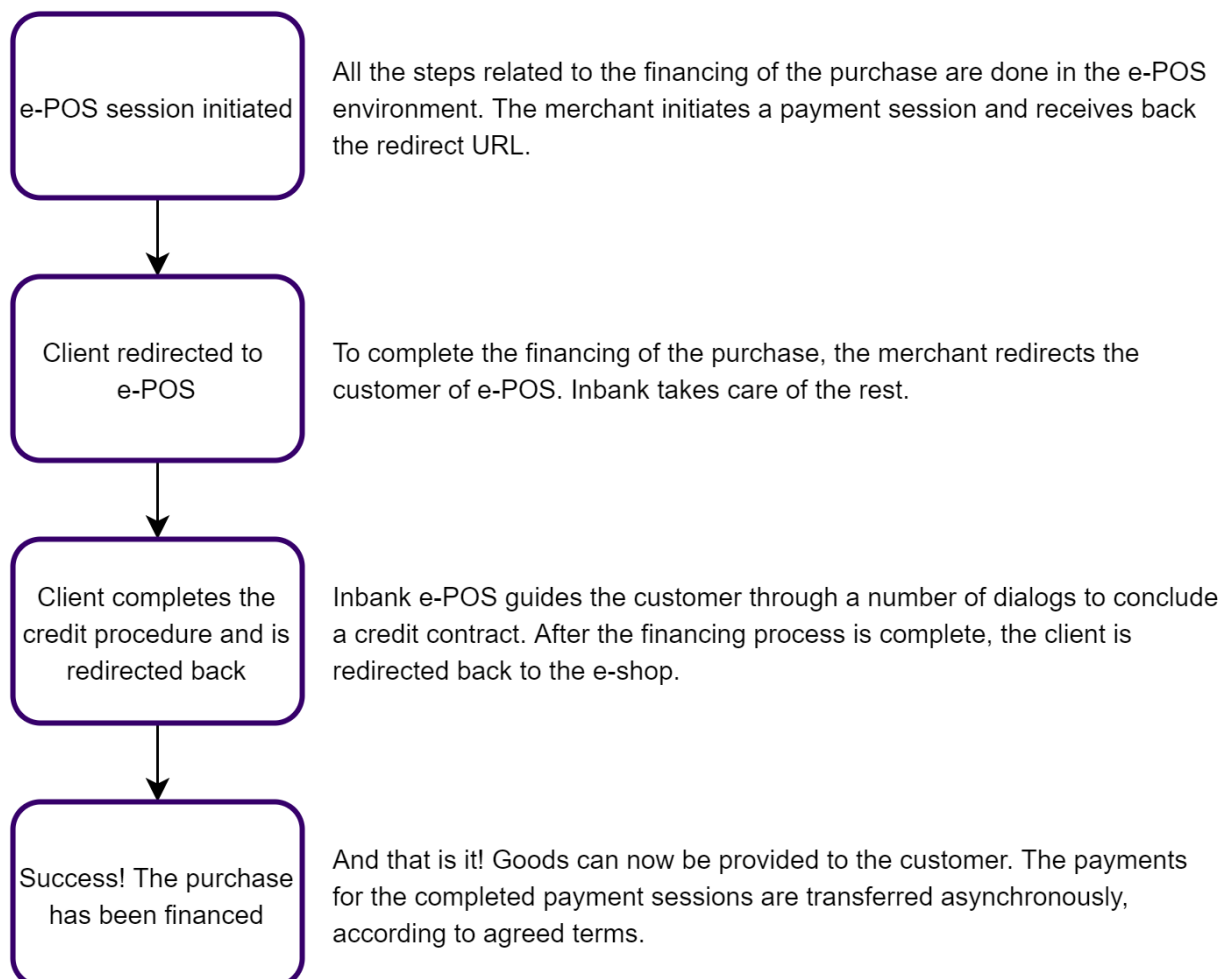| Version	Date	Updates
2.01	16.04.2020	Initial document
2.02	06.05.2020	Contact email address for questions regarding integration has been changed to integration@inbank.ee
2.03	29.06.2020	- Added information about contract statuses. - Added diagrams which depict the flow of API requests. - Updated information regarding the SMS simulator available for signing testing. - In the `POST /pos_session` request the `township` parameter is now used to indicate the city in the customer's address.
2.04	04.08.2020	- Added information about additional callbacks in case of Indivy payment product. - Updated information regarding the SMS simulator available for signing testing. - Added notes that the `GET /contracts` request is not needed when integrating with Indivy.
2.05	16.09.2020	Updated the integration process. `GET /pos_session/:uuid` and `POST /merchant` requests are no longer required to start a payment session in e-POS and have been removed from this document.
2.06	20.10.2020	Added possible purchase item types for the `POST /pos_session` endpoint.
2.07	22.01.2021	Added information on the amounts used in the demo environment for predefined decisions.
2.08	18.02.2021	Added information about the `valid_until` parameter added to `POST /pos_sessions`.
2.09	15.09.2021	Added a note to the API Requests section that Inbank payment methods should be available only for cart values that are within the price range agreed with Inbank.
2.10	10.03.2022	Updated information related to SMS messages used in the demo environment.

Flow Overview

In general, the flow looks like this:

e-POS session initiated	All the steps related to the financing of the purchase are done in the e-POS environment. The merchant initiates a payment session and receives back the redirect URL.
Client redirected to e-POS	To complete the financing of the purchase, the merchant redirects the customer of e-POS. Inbank takes care of the rest.
Client completes the credit procedure and is redirected back	Inbank e-POS guides the customer through a number of dialogs to conclude a credit contract. After the financing process is complete, the client is redirected back to the e-shop.
Success! The purchase has been financed	And that is it! Goods can now be provided to the customer. The payments for the completed payment sessions are transferred asynchronously, according to agreed terms.

Inbank also sends server-to-server notification messages to ensure delivery of information about the payment session even if the customer does not return to the e-shop.

Inbank will provide you with everything you need to start using our Partner API. This includes the necessary keys, product configuration, etc. For any questions regarding the integration process, contact Inbank at integration@inbank.ee.

Demo Environment

Inbank provides a separate environment for development and integration testing. The demo environment remains available during the later life cycle of our cooperation, after the integration on production environment has been launched. The demo and production environments are different, each having individual data sets.

Note that the access credentials and product codes are different in the production environment. You will be provided production specific information later on.

For testing purposes, the demo environment returns preconfigured decisions:
- For Inbank Hire Purchase Estonia and Indivy Pay in parts Estonia payment products, positive decision is given for amounts 0 - 500, 1001 - 3000, 15 000 - 16 000.
- For the Indivy payment product, a positive decision is given for amounts 0 - 100, 150 - …

The credit application process may include an OTP code exchange via SMS. The demo environments do not send out SMS messages. If you are testing Indivy or Indivy Pay in parts payment products, the SMS message is hardcoded to value 0000.

Guidelines for Integration Implementation

API Connectivity

Before you can initiate a session, Partner API connectivity must be configured.

Inbank will provide you an API key, used for authentication, and a unique identifier of your shop, required for building API URLs (for example POST /shops/**your_shop_uuid**/pos_sessions). **The keys should remain private at all times.**

Authentication

The authentication process consists of the following two steps:
1. Merchant places the API key in the Authorization header of the request.
2. API server verifies the API key authenticity.

Authorization Header

Authorization header must have the "Bearer" scheme and value of your API key, for example:
Authorization: Bearer e93174d3b9158a01c861c65fab0e7f96

In case of unsuccessful authorization, the system will return the following response:

HTTP code	Description
401	Unauthorized

```
{
    "error": [
        "unauthorized"
    ]
}
```

Content-Type

The HTTP header Content-Type application/json is expected in all requests, unless otherwise specified in the endpoint description. Example:
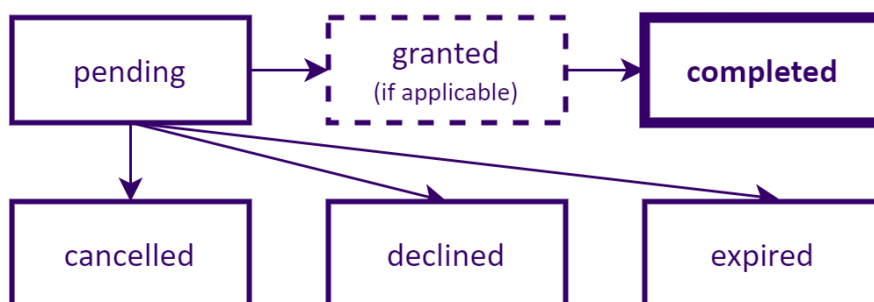
Content-Type: application/json

URLs

Environment	API	e-POS	Partner Portal
Test	https://demo-api.inbank.ee/partner/v2/	https://demo-epos.inbank.ee	https://demo-partner.inbank.ee
Production	https://api.inbank.ee/partner/v2/	https://epos.inbank.ee	https://partner.inbank.ee

Payment Session State Model

For the easiest integration we have designed the session status model to be similar to other payment channels that the e-shop integrates with.

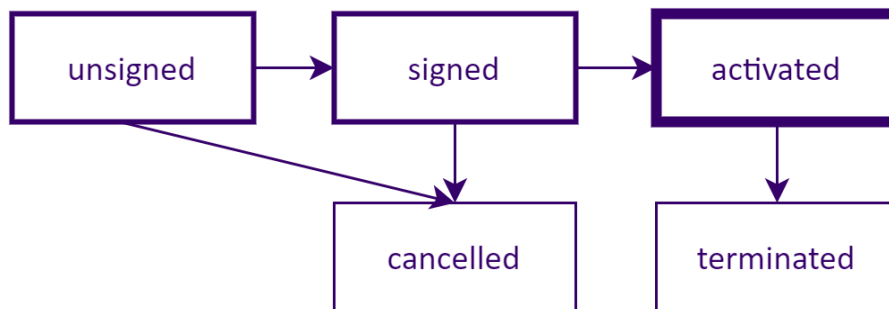| Status	Description
pending	A session is created; Credit application may be or not be in progress; Positive but not accepted credit decisions also remain in this status until they expire.
cancelled	The customer has cancelled the process.
granted	Credit has been granted to the customer, there are no obstacles from the Inbank side for sales completion. The process is now waiting for merchant's approval, if configured so. If the flow is configured not to wait for merchant's approval, this state may be omitted (see note below).
completed	This is the target state: credit contract between customer and Inbank has been activated, merchant is liable for the delivery of goods/services.
declined	Credit was declined by Inbank.
expired	The session was not completed during the defined time period.

The integration flow can be configured to require a final merchant-side confirmation step, before the credit application process is completed. This is somewhat similar to the credit card flows where the amount is first reserved on the credit card account (transaction is approved), and is later 'captured' after the merchant has completed the transaction.

This may be handy if the stock is limited and the merchant does not allocate stock items before it is ensured that the customer can get the credit. If the merchant does not send the final approval (i.e. items are out of stock, order can not be completed), the granted credit is not completed.

Credit Contract State Model

Inbank will send callbacks about changes to the credit contract status. Contracts can have the following statuses:

Status	Description
unsigned	A contract has been created, but has not yet been signed by the customer and/or Inbank.
signed	The contract has been signed by both the customer and Inbank. For the flow which includes merchant approval, this state indicates that the credit has been granted by Inbank and the system is now awaiting approval from the partner to activate the contract.
activated	This is the target state: credit contract between customer and Inbank has been activated, merchant is liable for the delivery of goods/services.
cancelled	The credit contract has been cancelled. This state applies only to contracts which previously were 'unsigned' or 'signed'. For the flow which includes merchant approval, 'signed' contracts get the status 'cancelled' when the merchant has not approved the contract.
terminated	An existing credit contract has been terminated. This state can only be applied to contracts which previously were 'activated'.

Callbacks

When initiating the payment session in Inbank Partner API the e-shop should provide 3 URLs:
- **redirect URL** - the URL to which the customer's browser will be redirected back after the customer has completed the credit application dialog. Regardless of the outcome of the credit application process, the customer is redirected here, even if the credit is not granted.
- **cancel URL** - the URL to which the customer's browser is redirected if the customer deliberately chooses to cancel the credit application dialog.
- **callback URL** - the URL to which Inbank will send server-to-server callback notifications on session status change events.

Inbank sends callbacks about the following state transition events:
- Credit application is cancelled
- Credit application is denied
- Contract is cancelled
- Contract is granted (applicable if the flow requires merchant approval of credit contracts)
- Contract is activated (state indicating that the financing of the purchase has been completed)

If you are integrating with Inbank's Indivy payment product, there can be cases when a customer already has an active credit contract and the new purchase is added to it. In this case, the following callbacks will be sent:
- Purchase is rejected (credit for the purchase has been declined by Inbank, the payment session status becomes Declined)
- Purchase is cancelled (the customer has cancelled the financing process, the payment session status becomes Cancelled)
- Purchase is activated (financing for the purchase has been successfully provided, the payment session status becomes Completed)

Once the financing process is finalized, Inbank will send two callbacks, both with the same structure and content:
- First one is performed on `redirect_url`, via the customer's browser.
- Second one is server-to-server callback which ensures that the merchant receives the callback. It is done on `callback_url`.

Note that the first callback may not arrive if the customer does not press the "**back to merchant**" button, or if there are connectivity or technical problems at the customer's device/browser. Thus there is no guarantee that the first callback will arrive, or which one of the two callbacks will arrive first. Callback requests are lightweight triggers for initiating activities on the merchant side. They contain only minimal information.

To avoid processing accidental or malicious traffic to callback endpoints, the handlers should first verify the authenticity of the request. For more details, see the Callback authenticity validation chapter.

E-shop should process the incoming messages, at a minimum, in the following way:
- Validate the authenticity of the request, to avoid further processing of invalid traffic.
- Look up the `pos_session` identifier either from the incoming message, or from the internal database as it was persisted when the session was initiated.
- Inspect `pos_session` status and process the order payment status based on the `pos_session` state. If needed, you can also check the purchase reference.
- Redirect the user to the respective dialog, i.e. the "payment complete" page.

Request Structure

Both of the callbacks are sent as http POST requests,
("Content-Type" => "application/x-www-form-urlencoded"). The POST form has the following structure:

Parameter	Example value	Description
message	`%7B%22uuid%22%3A%22e4b5b81a-6d99-4a78-bd17-46d19968eb7f%22%2C%22status%22%3A%22pending%22%2C%22purchase_reference%22%3A%22Id+%231%22%7D`	URL-encoded JSON structure containing information about the pos_session. For more details, see the Callback message content chapter.
hmac	`c196e985640a6291723dc2717d264f82e70126c34b107f3be5b22201cb147c98b9709f5184a7f2fe82684d6086eee07df8a46c28fc0edfdd14fd306579244664`	HMAC value. For more details, see HMAC calculation logic described in the Callback authenticity chapter.
timestamp	`1549411200`	Current Unix timestamp at issuing server. See https://en.wikipedia.org/wiki/Unix_time for more details.

Callback Request Example

Request header
`{"Content-Type":"application/x-www-form-urlencoded"}`

Request body
`message=%7B%22uuid%22%3A%223241a6d5-051b-415b-afc7-0a5aad115fcc%22%2C%22status%22%3A%22cancelled%22%2C%22purchase_reference%22%3A%221234%22%7D&hmac=4c4686db2aac832dd2e001fdc02e2b4021dc5e49c064552215dab2ca9c5649435562bc60e96b812ca8ea40223f500ced9c257541b43ab7fb482067c8bae7a963&timestamp=1553072069`

Callback Message Content

The message contains minimal information, it is meant as a trigger to obtaining more detailed information over Partner API. The message body contains:

- `uuid` - POS session UUID.
- *status* - status of the POS session at the moment of message dispatch. For more details, see the State model chapter.
- *purchase_reference* - merchant side reference, i.e. order ID. For more details, see the Session initiation chapter.

Callback Authenticity Validation

We use message authenticity hash (HMAC) transported within the POST request form field "hmac". To validate the message authenticity you need to calculate the verifying HMAC based on data from the request and your secret api_key, and compare the calculated HMAC with the HMAC value passed in the request.

Verifying HMAC is calculated as SHA512 HMAC, over the 'timestamp' and 'message' from the request, concatenated with "." delimiter.
Your shop API key is used as HMAC secret.

Pseudocode for example verifying HMAC calculation:
```
key = your_api_key;
req_timestamp = request[timestamp];
req_message = request[message];
req_data = req_timestamp+'.'+req_message;
v_hmac = hmac("sha512", key, req_data);
```

JavaScript example (Postman):
```
key = your_api_key;
req_timestamp = decodeURIComponent(request[timestamp]);
req_message = request[message];
req_data = req_timestamp + '.' + req_message;
v_hmac = CryptoJS.HmacSHA512(req_data, key);
```

PHP example:
```
$key = $settings->api_key;
$req_timestamp = $_POST['timestamp'];
$req_message = stripslashes($_POST['message']);
$v_hmac = hash_hmac('sha512', $req_timestamp . '.' . $req_message, $key);
```

API Requests

This section lists the API request required for the integration with the Inbank e-POS system. The following pages contain charts with demonstration of the request sequence. The enlisted API requests are used in the following way:

1. The shop retrieves a primary credit calculation using the `POST /calculations` request. The response includes an approximate monthly payment based on the credit amount and period. The final conditions will be presented in e-POS after the customer submits an application.

   Note that Inbank payment methods should be available only for cart values that are within the price range agreed with Inbank. If you would like to receive the price range and other details of your Inbank product over API, please contact integration@inbank.ee.

2. The e-shop initiates a payment session using the `POST /pos_sessions` endpoint. The request includes merchant domain name as one of the parameters. The `redirect_url` from the response indicates the link to which the client is redirected to complete the financing process.

3. The e-shop redirects the client to the e-POS environment. In e-POS customers are guided through a number of dialogs to complete the financing of the purchase. After the e-POS dialogs, customers are redirected back to the e-shop. The `return_url` is the one the e-shop included in the `POST /pos_sessions` request.

4. If the flow is configured to request merchant approval before contract activation, the e-shop waits for the callback indicating that the payment session received the status `granted`. At this point, the e-shop retrieves the identifier of the contract using the `GET /pos_sessions` request. After that, the merchant can either approve the credit contract, using `POST /:contract_uuid/merchant_approval` request, or cancel it, using the `POST /:contract_uuid/cancel`. The following step is necessary only if the contract was approved.

5. Once the e-shop receives the callback indicating that the payment session received the status `completed`, the e-shop needs to check the contract status. First, the e-shop retrieves the identifier of the contract using the `GET /pos_sessions` request. Retrieving the contract identifier again is not required if it was previously done to approve the contract. Then the e-shop checks the status of the credit contract using the `GET /contracts` request. If the contract received status `activated`, the financing of the purchase has been successful. **Note that this step should not be performed if you are integrating with the Indivy payment product.**

API Request Flow

The chart demonstrates the sequence in which the API requests should be applied to successfully initiate the payment session, redirect the customer to e-POS and later check the credit contract status to confirm that the financing has been successful.
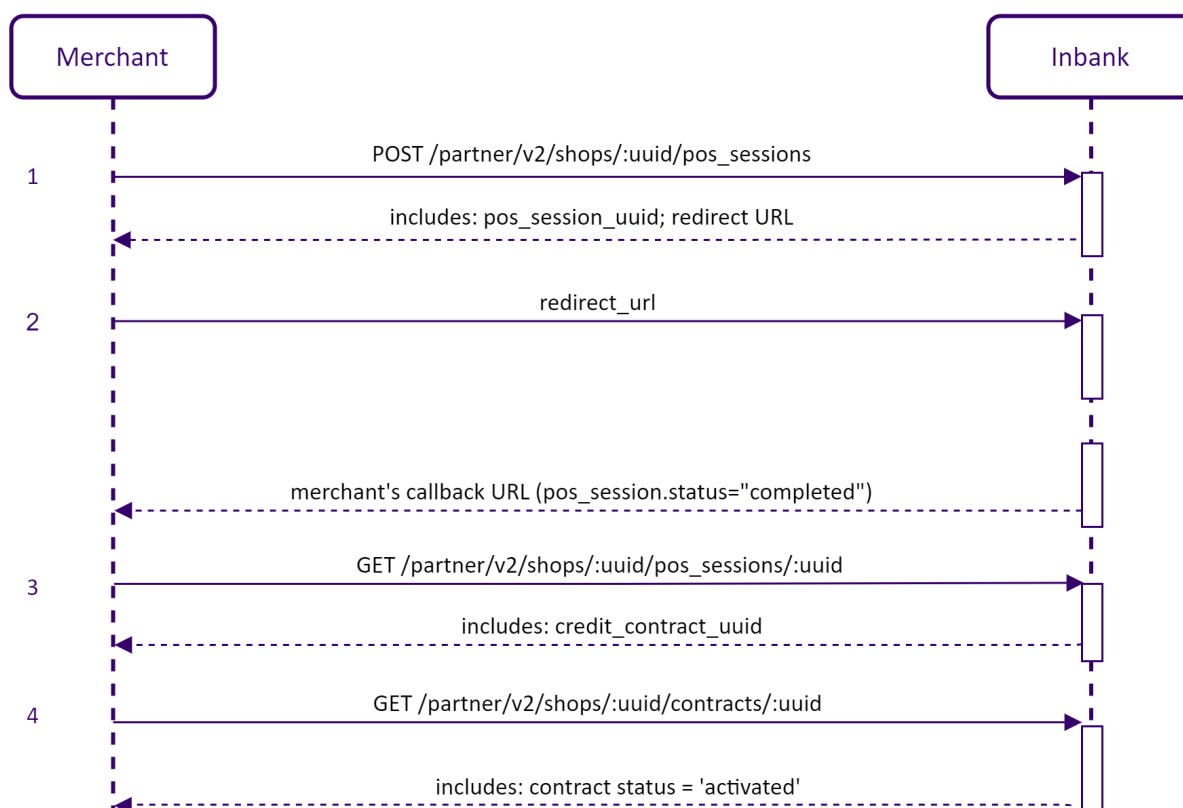
Please note that if you are integrating with the Indivy payment product, you do not need to perform steps 3 and 4.

API Request Flow with Merchant Approval

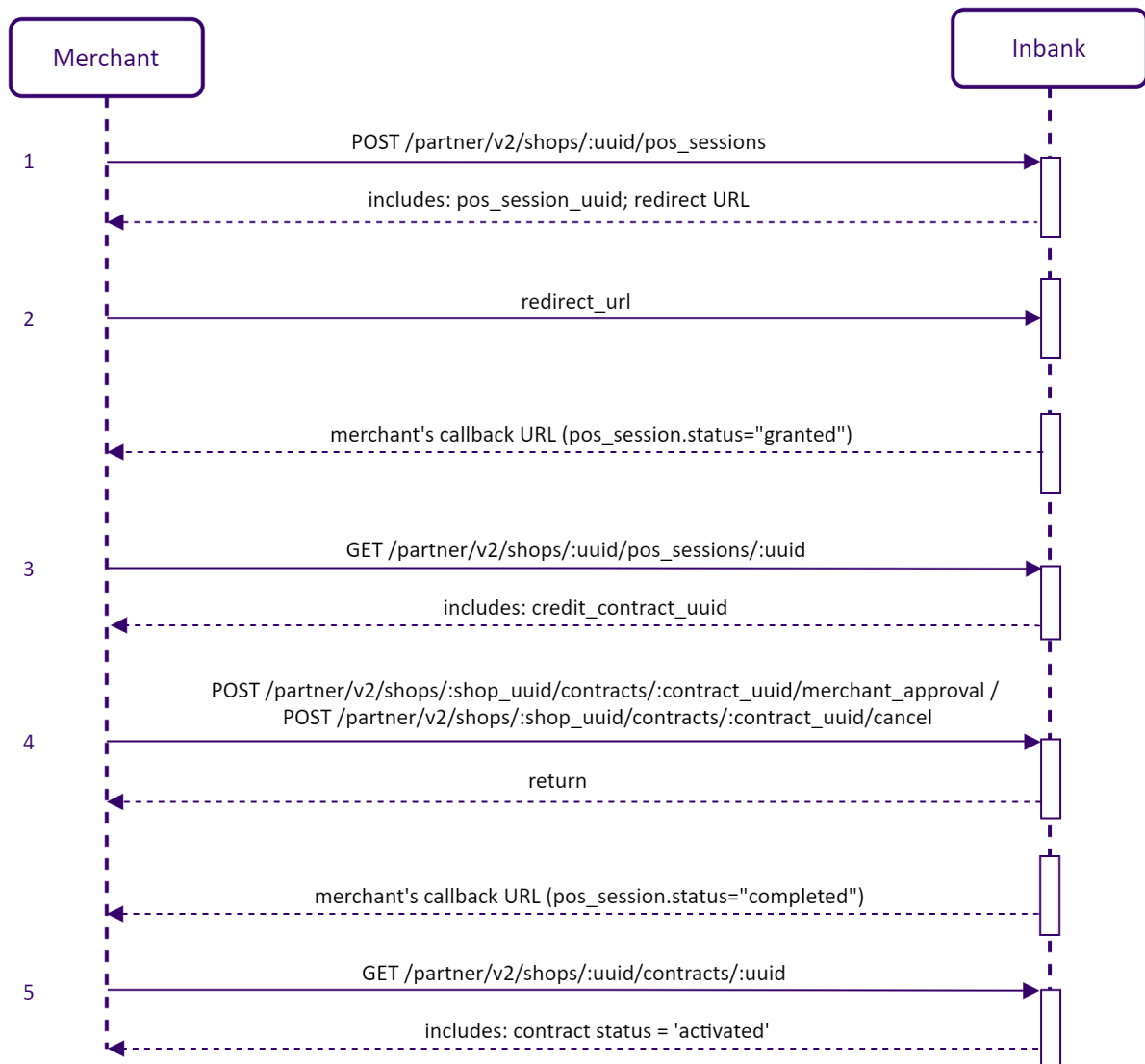The chart below applies to cases when the flow requires merchant approval prior to contract activation. The chart demonstrates the sequence in which the API requests should be applied to successfully initiate the payment session, redirect the customer to e-POS and later check the credit contract status to confirm that the financing has been successful.

Please note that if you are integrating with the Indivy payment product, you do not need to perform step 5.

Calculator

`POST /shops/:uuid/calculations`

To get a credit calculation from Inbank, use the `POST /shops/:uuid/calculations` request.

Note that this request returns the preliminary non-personalized credit conditions. The final conditions will be presented after the customer submits a credit application and receives a positive decision.

The following request body parameters are accepted:

Name	Required	Type	Description
product_code	Yes	String	Reference to product code.
amount	Yes	Numeric	Total financing amount. If down payment is used, then the amount includes the down payment amount. Minimum and maximum amount depends on product setup.
period	Yes	Numeric	Number of months for repayments; allowed values depend on product setup.
down_payment_amount	No	Numeric	Down payment amount.
currency	No	String	Currency code in uppercase. Available option is EUR.
response_level	No	String	Indication of how detailed the returned response should be. Available options: ● simple (default) - basic calculation data ● advanced - more parameters for the same calculation ● payment_schedule - includes all payment schedule rows It is recommended to use the minimum response level that is needed for a specific calculation, e.g. if only monthly payment is needed, then use "simple".

Request body example

```
{
    "product_code": "product_code_here",
    "amount": 2000,
    "period": "12",
    "down_payment_amount": "0",
    "currency": "EUR",
    "response_level": "simple"
}
```

Response example

```
{
    "product_code": "product_code_here",
    "amount": "2000.0",
    "period": 12,
    "down_payment_amount": "0.0",
    "payment_day": 10,
    "response_level": "simple",
    "currency": "EUR",
    "payment_amount_monthly": "177.86",
    "interest_rate_annual": "0.0899",
    "credit_cost_rate_annual": "0.1287",
    "total_cost": "2134.26",
    "total_cost_of_credit": "134.26"
}
```

Session Initiation

`POST /shops/:uuid/pos_sessions`

To start a payment session in Inbank e-POS, use the `POST /shops/:uuid/pos_session`. The response includes the identifier of the payment session - `pos_session_uuid` and the URL to which the customer is to be redirected - `redirect_url`.

For your convenience, we have listed the [minimal data set](#) which needs to be passed to Inbank.

The request has following attributes:

Name	Required	Type	Description
General			
product_code	True	String	Reference to product code.
salesperson_reference	False	String	Identifier of the salesperson as it should appear on reports.
total_amount	True	Numeric	Total amount to be paid by the customer (should include shipping, packaging, discount, etc).
currency	True	String	Currency code in uppercase. Available option is EUR.
locale	True	String	Language - country codes, for example "et-ET".
valid_until	False	String	The point in time when the payment session is to expire. E.g. `2021-02-17T11:10:00+02:00`. After this time, the customer will not be able to complete the financing procedure.
user_ip	False	String	Customer browser IP address as seen by e-shop.
Customer data			

customer_data	True*	JSON	Data used to pre-fill credit application fields regarding the customer. Allowed keys: ● identity_code* ● first_name* ● last_name* ● gender - options: m (male), f (female).
customer_contact_data	True*	JSON	Data used to pre-fill credit application fields regarding customer contact information. Allowed keys: ● email* ● mobile*
customer_address_data	False	JSON	Data used to pre-fill credit application fields regarding customer address. Allowed keys: ● country ● zip_code ● street ● parish ● house ● apartment ● township ● county
Purchase			
purchase[purchase_reference]	True	String	Unique identifier of the shopping cart or order in the e-shop (i.e. order ID).
purchase[description]	False	String	Textual description of the order or shopping cart, meaningful for the customer.
purchase[additional_details]	False	String	Any additional details about the order that are not covered with predefined parameters.
Purchase merchant details			
purchase[merchant]	True*	Array	Array of items containing information about the merchant supplying the purchase.

purchase[merchant][merchant_name]	False	String	Name of the merchant who is associated with a particular purchase.
purchase[merchant][merchant_registry_code]	False	String	Merchant's registry code, consisting of eight to eleven digits.
purchase[merchant][merchant_order_reference]	False	String	Order reference of the merchant.
purchase[merchant][merchant_domain_name]	True	String	URL of the merchant. This is the URL of the partner website from which purchases are done.
purchase[merchant][merchant_mcc_code]	False	String	4-digit merchant category code listed in ISO 18245 for retail financial services. This code is used to classify a business by the type of goods or services it provides.

Purchase details

purchase[items]	False	Array	Array of items in the order/shopping cart.
purchase[items][n][item_reference]	False	String	Unique item ID in the e-shop, inventory code etc.
purchase[items][n][type]	False	String	Item type. Some types require additional information/validations. Available options: product, service, vehicle.
purchase[items][n][description]	False	String	Textual name of the item.
purchase[items][n][quantity]	False	Integer	Quantity of items in this row.
purchase[items][n][amount]	False	Number	Price of item(s) on this row (row sum).
purchase[items][n]serial_number]	False	String	Serial number of the purchased item.
purchase[items][n]url]	False	String	Link to the purchased item.

purchase[items][n]image_url]	False	String	Link to an image of the purchased item.

Additional details for vehicle purchases

purchase[items][n]vehicle_vin]	False	String	Vehicle identification number of the purchased vehicle.
purchase[items][n]vehicle_license_plate]	False	String	License plate of the purchased vehicle.
purchase[items][n]vehicle_make]	False	String	Brand of the purchased vehicle.
purchase[items][n]vehicle_model]	False	String	Model of the purchased vehicle.
purchase[items][n]vehicle_registration_date]	False	String	Registration date of the purchased vehicle.

Credit application data

credit_application_data	False	JSON	Credit application related attributes that a merchant can pass, and Inbank can use to pre-fill applications in e-POS. Allowed keys: - period - payment_day - monthly_income - dependants - monthly_household_costs - income_source

Partner URLs

partner_urls[return_url]	True	String	Partner defined URL to which the customer should be redirected after credit application dialog.
partner_urls[cancel_url]	True	String	Partner defined URL to which the customer should be redirected in case they deliberately cancelled the credit application dialog.

partner_urls[callback_url]	True	String	URL for sending server-to-server callback messages about session status changes.

Integration data

integration_info	False	JSON	Information about the technical environment of the partner side integration, e.g.: "ecom_platform":"string", "module":"string"

Additional fields

additional_data	False	JSON	Container for additional data that e-shops can pass to e-POS dialogs (pre-filling forms). Any keys are allowed.

* The `customer_data`, `customer_contact_data` and `merchant` objects and parameters included in them are optional. A request that does not contain these objects will be processed correctly. However, if the body does contain these objects, Inbank will validate the parameters passed inside them. Therefore, if the request contains `customer_data`/`customer_contact_data`/`merchant` objects, their parameters become required.

Request body example

```
{
    "product_code": "car_loan",
    "total_amount": 9000,
    "currency": "EUR",
    "locale": "et-ET",
    "valid_until": "2021-02-17T11:10:00+02:00",
    "salesperson_reference": "REF120",
    "partner_urls": {
        "return_url": "https://campaign.inbank.ee/tmp/post.php?type=return",
        "cancel_url": "https://campaign.inbank.ee/tmp/post.php?type=cancel",
        "callback_url": "https://campaign.inbank.ee/tmp/post.php?type=callback"
    },
    "purchase": {
        "purchase_reference": "ORDER_000001",
        "description": "Description of ORDER_000001 order",
        "merchant": {
            "merchant_name": "Example OÜ",
            "merchant_registry_code": "12345678901",
            "merchant_order_reference": "T21093289",
            "merchant_domain_name": "wwww.example.com",
            "merchant_mcc_code": "5812"
        },
        "additional_details": {
```

```
            "description": "Purchase additional details"
        },
        "items": [
            {
                "item_reference": "000002",
                "type": "vehicle",
                "description": "audi A6",
                "quantity": "8",
                "amount": "4800",
                "serial_number": "SN_000001",
                "url": "https://en.wikipedia.org/wiki/Audi_A6",
                "image_url": "https://en.wikipedia.org/wiki/Audi_A6#
                            /media/File:2007_Audi_A6_(4F)_allroad_quattro_
                            3.0_TDI_station_wagon_02.jpg",
                "additional_details": {
                    "owner_amount": "1"
                },
                "vehicle_vin": "VIN000001",
                "vehicle_licence_plate": "111AAA",
                "vehicle_make": "Audi",
                "vehicle_model": "A6",
                "vehicle_registration_date": "01.08.2014"
            },
            {
                "item_reference": "000001",
                "type": "service",
                "quantity": "2",
                "amount": "4800"
            }
        ]
    },
    "user_ip": "192.128.00.01",
    "customer_data": {
        "first_name": "John",
        "last_name": "Smith"
    },
    "customer_contact_data": {
        "email": "john.smith@session.pos",
        "mobile": "51231412",
        "phone": "6123123"
    },
    "customer_address_data": {
            "type": "legal",
            "street": "PAE TEE",
            "country": "EE",
            "county": "HARJU MAAKOND",
            "zip_code": "03309",
            "house": "16",
            "township": "RAE KÜLA"
    },
    "credit_application_data": {
            "number": "8000000123",
            "salesperson_reference": "SP_REF_001"
```

```
        },
        "integration_info": {
              "ecom_platform": "magento",
         "module": "inbank-2.1.0",
         "extra_key_3": "#3"
        },
        "additional_data": {
              "key_1": "key1",
         "key_2": "key2",
         "key_3": "key3"
        }
}
```

Response example

```
{
    "uuid": "a1b1ec1f-1cd1-111b-1ed1",
    "status": "pending",
    "redirect_url": "https://staging-epos.inbank.ee/session/a8b5ec3f-1cd2-477b-9ed"
}
```

Response for a successful request will return:
- uuid - identifier of the POS session. Store this identifier at the order metadata for later usage.
- status - the status of the POS session.
- redirect_url - the link to which the client is to be redirected to complete the financing process in the e-POS environment

Minimal Data Set Example
The following sample contains a minimal set of parameters which are required for request processing.

```
{
    "product_code": "small_loan",
    "total_amount": 3000,
    "currency": "EUR",
    "locale": "et-ET",
    "partner_urls": {
        "return_url": "https://campaign.inbank.ee/tmp/post.php?type=return",
        "cancel_url": "https://campaign.inbank.ee/tmp/post.php?type=cancel",
        "callback_url": "https://campaign.inbank.ee/tmp/post.php?type=callback"
    },
    "purchase": {
        "purchase_reference": "ORDER_000001",
        "merchant": {
            "merchant_domain_name": "wwww.example.com"
        }
    }
}
```

Session Details

`GET /shops/:shop_uuid/pos_sessions/:pos_session_uuid`

When a user is redirected back to e-shop, or when a callback notification is received, the e-shop should make a `GET /shops/:shop_uuid/pos_sessions/:pos_session_uuid` request to inspect session details.

The response contains the `credit_contract_uuid` value which is used in the GET /contracts request to check the status of the contract. If the flow is configured to request merchant approval before credit contract activation, this value is also used in the POST /:contract_uuid/merchant_approval or the POST /:contract_uuid/cancel request, to either approve or cancel the credit contract.

Response example

```
{
    "uuid": "7ed7fab8-316a-4f42-9a52-1e9c48a00000",
    "product_code": "car_loan",
    "total_amount": "2000.0",
    "currency": "EUR",
    "status": "pending",
    "salesperson_reference": "REF120",
    "locale": "et-ET",
    "user_ip": "192.000.00.00",
    "partner_urls": {
        "return_url": "https://test.inbank.ee/tmp/post.php?type=return",
        "cancel_url": "https://test.inbank.ee/tmp/post.php?type=cancel",
        "callback_url": "https://test.inbank.ee/tmp/post.php?type=callback"
    },
    "customer_data": {
        "last_name": "Smith",
        "first_name": "John"
    },
    "customer_contact_data": {
        "email": "john.smith@session.pos",
        "phone": "6123123",
        "mobile": "51231412"
    },
    "customer_address_data": {
        "city": "TALLINN",
        "house": "3B",
        "street": "RAUA",
        "country": "EE",
        "zip_code": "03309"
    },
    "credit_application_data": {
```

```json
        "number": "P89003030000",
        "salesperson_reference": "SP_REF_001"
    },
    "integration_info": {
        "module": "inbank-2.1.0",
        "extra_key_3": "#3",
        "ecom_platform": "magento"
    },
    "additional_data": {
        "key_1": "key-1",
        "key_2": "key-2",
    },
    "purchase": {
        "purchase_reference": "ORDER_000001",
        "description": "Description of ORDER_000001 order",
        "uuid": "25a81dc5-95ac-4087-b646-e31353000000",
        "additional_details": {
            "description": "Purchase additional details"
        },
        "items": [
            {
                "item_reference": "000001",
                "type": "vehicle",
                "description": "audi A6",
                "quantity": 8,
                "amount": "200.0",
                "serial_number": "SN_000001",
                "url": "https://en.wikipedia.org/wiki/Audi_A6",
                "image_url":
"https://en.wikipedia.org/wiki/Audi_A6#/media/File:2007_Audi_A6_(4F)_allroad_quat
tro_3.0_TDI_station_wagon_02.jpg",
                "additional_details": {
                    "owner_amount": "1"
                },
                "vehicle_vin": "VIN000000",
                "vehicle_licence_plate": "111AAA",
                "vehicle_make": "Audi",
                "vehicle_model": "A6",
                "vehicle_registration_date": "01.08.2014"
            }
        ],
        "created_at": "2020-02-21T13:31:01+01:00"
    },
    "created_at": "2020-02-21T13:31:01+01:00",
    "valid_until": "2020-02-28T13:31:01+01:00",
    "credit_application_uuid": null,
    "credit_contract_uuid": null
}
```

The response partly reflects the data that was submitted on session initiation, but also includes some important attributes to be used:

Name	Type	Description
status	String	Session status. Payment for the e-shop order can be considered successfully completed when the status = 'completed'. For more details, see the State model chapter.
credit_application_uuid	UUID	Identifier of the credit application in Inbank systems. This parameter appears in the response once the customer has submitted the application during dialogs. Example: b1904cd8-f5b0-4610-b87c-97a512d6125f
credit_contract_uuid	UUID	The unique identifier of the credit contract associated with the session. This identifier is used in the GET /:contract_uuid/contracts request to check the status of the contract or in the POST /merchant_approval or the POST /:contract_uuid/cancel request, to either approve or cancel the credit contract.

It is important to inspect the value of the status. If the status is completed, then from the e-shop order perspective it has been paid, and the goods can be shipped.

Contract Approval

`POST /partner/v2/shops/:shop_uuid/contracts/:contract_uuid/merchant_approval`

If the flow is configured to request merchant approval, the e-shop will receive the callback informing that the payment session has received status `granted`. This means that the credit has been approved by Inbank.

To approve the contract, the e-shop first needs to perform the [GET /pos_sessions](#) request, which, among other parameters, returns the `credit_contract_uuid`. This identifier can then be used to approve the credit contract.

The request does not require any parameters to be passed in its body.

Response

HTTP code	Description
204	No Content

Contract Cancellation

`POST /partner/v2/shops/:shop_uuid/contracts/:contract_uuid/cancel`

If the flow is configured to request merchant approval, the e-shop will receive the callback informing that the payment session has received status `granted`. This means that the credit has been approved by Inbank.

To cancel the contract, the e-shop first needs to perform the [GET /pos_sessions](#) request, which, among other parameters, returns the `credit_contract_uuid`. This identifier can then be used to cancel the credit contract.

The request does not require any parameters to be passed in its body.

Response

HTTP code	Description
204	No Content

Contract Details

Note that this request should not be used if you are integrating with the Indivy payment product.

`GET /partner/v2/shops/:shop_uuid/contracts/:contract_uuid`

Once the credit contract UUID has been retrieved via the [GET /pos_sessions](#) request, the e-shop can check the status of the credit contract using the `GET /partner/v2/shops/:shop_uuid/contracts/:contract_uuid` request. The response will include the `status` parameter. If the `status` is `activated`, the purchase has been successfully financed by Inbank and the purchase items can be forwarded to the customer.

Response example

```
{
    "contract": {
        "status": "unsigned",
        "termination_reason": null,
        "uuid": "11d1baeb-1da1-1c01-b111-12111211c1a1",
        "number": "89001350000",
        "payout_account_number": "EE1982484545379277458000000",
        "activated_at": null,
        "activator_name": null,
        "terminated_at": null,
        "product_code": "insurance_fin",
        "customer_signed": null,
        "rep_signed": null,
        "signed_at": null,
        "partner_approval_at": null,
        "customer_uuid": "40837f6d-0000-0000-0000-59a5b1efedd8",
        "identification_satisfied": true
    }
}
```