mollie docs

Create payment

Payments API v2

POST `https://api.mollie.com/v2/pay…`

Authentication: `API keys` `Organization access tokens` `App access tokens`

Payment creation is elemental to the Mollie API: this is where most payment implementations start off.

Once you have created a payment, you should redirect your customer to the URL in the `_links.checkout` property from the response.

To wrap your head around the payment process, an explanation and flow charts can be found in the Accepting payments guide.

Note

Optional parameters are accepted for certain payment methods.

Parameters

`amount` amount object **REQUIRED**

The amount that you want to charge, e.g. `{"currency":"EUR", "value":"1000.00"}` if you would want to charge €1000.00.

You can find the minimum and maximum amounts per payment method in our help center. Additionally, they can be retrieved using Get payment method.

`currency` string **REQUIRED**

An ISO 4217 currency code. The currencies supported depend on the payment methods that are enabled on your account.

`value` string **REQUIRED**

A string containing the exact amount you want to charge in the given currency. Make sure to send the right amount of decimals and omit the thousands separator. Non-string values are not accepted.

`description` string **REQUIRED**

The description of the payment you are creating. This will be shown to your customer on their card or bank statement when possible. We truncate the description automatically according to the limits of the used payment method. The description is also visible in any exports you generate.

We recommend you use a unique identifier so that you can always link the payment to the order in your back office. This is particularly useful for bookkeeping.

The maximum length of the description field differs per payment method, with the absolute maximum being 255 characters. The API will not reject strings longer than the maximum length but it will truncate them to fit.

`redirectUrl` string **REQUIRED**

The URL your customer will be redirected to after the payment process.

It could make sense for the `redirectUrl` to contain a unique identifier – like your order ID – so you can show the right page referencing the order when your customer returns.

The parameter can be omitted for recurring payments (`sequenceType: recurring`) and for Apple Pay payments with an `applePayPaymentToken`.

`cancelUrl` string **OPTIONAL**

The URL your consumer will be redirected to when the consumer explicitly cancels the payment. If this URL is not provided, the consumer will be redirected to the `redirectUrl` instead — see above.

Mollie will always give you status updates via webhooks, including for the `canceled` status. This parameter is therefore entirely optional, but can be useful when implementing a dedicated consumer-facing flow to handle payment cancellations.

The parameter can be omitted for recurring payments (`sequenceType: recurring`) and for Apple Pay payments with an `applePayPaymentToken`.

`webhookUrl` string **OPTIONAL**

Set the webhook URL, where we will send payment status updates to.

The `webhookUrl` is optional, but without a webhook you will miss out on important status changes to your payment.

The `webhookUrl` must be reachable from Mollie's point of view, so you cannot use `localhost`. If you want to use webhook during development on `localhost`, you must use a tool like ngrok to have the webhooks delivered to your local machine.

`lines` array **OPTIONAL** **BETA**

The lines for the payment. Each line contains details such as a description of the item ordered, its price et cetera.

All lines must have the same currency as the payment. You cannot mix currencies within a single payment.

`type` string **OPTIONAL**

The type of product bought, for example, a physical or a digital product.

Possible values: `physical` `discount` `digital` `shipping_fee` `store_credit` `gift_card` `surcharge`

`description` string **REQUIRED**

A description of the line item, for example *LEGO 4440 Forest Police Station*.

`quantity` int **REQUIRED**

The number of items in the line.

`quantityUnit` string **OPTIONAL**

The unit for the quantity, for example *pcs*, *kg*, *cm*, etc.

`unitPrice` amount object **REQUIRED**

The price of a single item including VAT.

For example: `{"currency":"EUR", "value":"89.00"}` if the box of LEGO costs €89.00 each.

Should be negative if `type` is `discount`, `store_credit` or `gift_card`, or zero in case of a free item.

`currency` string **REQUIRED**

An ISO 4217 currency code.

`value` string **REQUIRED**

A string containing the exact amount in the given currency.

`discountAmount` amount object **OPTIONAL**

Any discounts applied to the line. For example, if you have a two-for-one sale, you should pass the amount discounted as a positive amount.

For example: `{"currency":"EUR", "value":"10.00"}` if you want to give a €10.00 discount on this line.

`currency` string **REQUIRED**

An ISO 4217 currency code.

`value` string **REQUIRED**

A string containing the exact amount in the given currency.

`totalAmount` amount object **REQUIRED**

The total amount of the line, including VAT and discounts. For example: `{"currency":"EUR", "value":"168.00"}` if the total amount of this line is €168.00.

Should match the following formula: `(unitPrice × quantity) - discountAmount`.

Adding `totalAmount` values of all lines together should result in the same amount as the `amount` top level property.

`currency` string **REQUIRED**

An ISO 4217 currency code.

`value` string **REQUIRED**

A string containing the exact amount in the given currency.

`vatRate` string **OPTIONAL**

The VAT rate applied to the line, for example `"21.00"` for 21%. The `vatRate` should be passed as a string and not as a float to ensure the correct number of decimals are passed.

`vatAmount` amount object **OPTIONAL**

The amount of value-added tax on the line. The `totalAmount` field includes VAT, so the `vatAmount` can be calculated with the formula `totalAmount × (vatRate / (100 + vatRate))`.

Any deviations from this will result in an error.

For example, for a `totalAmount` of SEK100.00 with a 25.00% VAT rate you would get a VAT amount of `100.00 × (25 / 125)` = SEK20.00. The amount should be passed as an amount object, so: `{"currency":"SEK", "value":"20.00"}`.

`currency` string **REQUIRED**

An ISO 4217 currency code.

`value` string **REQUIRED**

A string containing the exact amount in the given currency.

`sku` string **OPTIONAL**

The SKU, EAN, ISBN or UPC of the product sold. The maximum character length is 64.

`imageUrl` string **OPTIONAL**

A link pointing to an image of the product sold.

`productUrl` string **OPTIONAL**

A link pointing to the product page in your web shop of the product sold.

`locale` string **OPTIONAL**

Allows you to preset the language to be used in the hosted payment pages shown to the consumer. Setting a locale is highly recommended and will greatly improve your conversion rate. When this parameter is omitted, the browser language will be used instead if supported by the payment method. You can provide any `xx_XX` format ISO 15897 locale, but our hosted payment pages currently only support the following languages:

Possible values: `en_US` `en_GB` `nl_NL` `nl_BE` `fr_FR` `fr_BE` `de_DE` `de_AT` `de_CH` `es_ES` `ca_ES` `pt_PT` `it_IT` `nb_NO` `sv_SE` `fi_FI` `da_DK` `is_IS` `hu_HU` `pl_PL` `lv_LV` `lt_LT`

`method` string|array **OPTIONAL**

Normally, a payment method screen is shown. However, when using this parameter, you can choose a specific payment method and your customer will skip the selection screen and is sent directly to the chosen payment method. The parameter enables you to fully integrate the payment method selection into your website.

You can also specify the methods in an array. By doing so we will still show the payment method selection screen but will only show the methods specified in the array. For example, you can use this functionality to only show payment methods from a specific country to your customer `['bancontact', 'belfius']`.

Possible values: `applepay` `bancontact` `banktransfer` `belfius` `creditcard` `directdebit` `eps` `giftcard` `giropay` `ideal` `kbc` `mybank` `paypal` `paysafecard` `przelewy24` `sofort` `twint`

Note

If you are looking to create payments with the Klarna Pay now, Klarna Pay later, Klarna Slice it, in3 or voucher payment methods, use Create order instead.

`restrictPaymentMethodsToCountry` string **OPTIONAL** Show details ⌄

`metadata` mixed **OPTIONAL** Show details ⌄

Parameters for recurring payments

Recurring payments are created through the Payments API by providing a `sequenceType`. For the `recurring` sequence type, you have to provide a `customerId` to indicate which account you want to charge. You can charge a card by providing a `mandateId`. See our guide on Recurring for more information.

`sequenceType` string **REQUIRED FOR RECURRING** Show details ⌄

`customerId` string **REQUIRED** Show details ⌄

`mandateId` string **OPTIONAL** Show details ⌄

Parameters for pre-authorized payments

Pre-authorized payments are created through the Payments API by providing a `captureMode`. With the `manual` capture mode, you decide when to capture the funds. See our guide on Placing a hold for a payment for more information.

`captureMode` string **OPTIONAL**

Indicates whether the capture will be scheduled automatically or not. Set to `manual` to capture the payment manually using the Create capture endpoint.

Set to `automatic` by default, which indicates the payment will be captured automatically, without having to separately request it. Setting `automatic` without a `captureDelay` will result in a regular payment.

Possible values: `automatic` `manual`

`captureDelay` string **OPTIONAL**

Interval to wait before the payment is captured, for example `8 hours` or `2 days`. In order to schedule an automatic capture, the `captureMode` must be set to either `automatic` or be omitted.

Possible values: `... hours` `... days`

Note

The maximum delay is 7 days (`168 hours`).

Payment method-specific parameters

If you specify the `method` parameter, optional parameters may be available for the payment method. If no method is specified, you can still send the optional parameters and we will apply them when the consumer selects the relevant payment method.

Bank transfer ▾

Bank transfer

`billingEmail` string **OPTIONAL**

Consumer's email address, to automatically send the bank transfer details to. **Note:** the payment instructions will will be sent immediately when creating the payment. If you do not specify the `locale` parameter, the email will be sent in English, as we haven't yet been able to detect the consumer's browser language.

`dueDate` string **OPTIONAL**

The date the payment should expire, in `YYYY-MM-DD` format. **Note:** the minimum date is tomorrow and the maximum date is 100 days after tomorrow.

After you created the payment, you can still update the `dueDate` via Update payment.

`locale` string **OPTIONAL**

For bank transfer payments specifically, the locale will determine the target bank account the customer has to transfer the money to. We have dedicated bank accounts for Belgium, Germany, and The Netherlands. Having the customer use a local bank account greatly increases the conversion and speed of payment.

Possible values: `en_US` `en_GB` `nl_NL` `nl_BE` `fr_FR` `fr_BE` `de_DE` `de_AT` `de_CH` `es_ES` `ca_ES` `pt_PT` `it_IT` `nb_NO` `sv_SE` `fi_FI` `da_DK` `is_IS` `hu_HU` `pl_PL` `lv_LV` `lt_LT`

Access token parameters

If you are using organization access tokens or are creating an OAuth app, you have to specify which profile you are creating the payment for using the `profileId` parameter. Organizations can have multiple profiles for each of their websites. See Profiles API for more information.

For these authentication methods the optional `testmode` parameter is available as well to enable test mode.

`profileId` string **REQUIRED FOR ACCESS TOKENS** Show details ⌄

`testmode` boolean **OPTIONAL** Show details ⌄

Mollie Connect parameters

With Mollie Connect you can charge fees on payments that are processed through your app, either by defining an *application fee* or by *splitting the payment*. To learn more about the difference, refer to the Mollie Connect overview.

`applicationFee` object **OPTIONAL** Show details ⌄

`routing` array **OPTIONAL** Show details ⌄

QR codes

To create a payment with a QR code embedded in the API response, explicitly set the payment method and call the API endpoint with an include request for `details.qrCode` in the query string:

```
POST    https://api.mollie.com/v2/payments?include=detail…
```

QR codes can be generated for iDEAL, Bancontact and bank transfer payments.

Refer to the Get payment reference to see what the API response looks like when the QR code is included.

Response

`201` `application/hal+json`

A payment object is returned, as described in Get payment.

Example

```
<?php
$mollie = new \Mollie\Api\MollieApiClient();
$mollie->setApiKey("test_dHar4XY7LxsDOtmnkVtjNVWXLSlXsM");
$payment = $mollie->payments->create([
      "amount" => [
            "currency" => "EUR",
            "value" => "10.00" // You must send the correct number o
      ],
      "description" => "Order #12345",
      "redirectUrl" => "https://webshop.example.org/order/12345/",
      "webhookUrl" => "https://webshop.example.org/payments/webhook/
      "metadata" => [
            "order_id" => "12345",
      ],
]);
```

Response

```
HTTP/1.1 201 Created
Content-Type: application/hal+json

{
    "resource": "payment",
    "id": "tr_7UhSN1zuXS",
    "mode": "test",
    "createdAt": "2018-03-20T09:13:37+00:00",
    "amount": {
        "value": "10.00",
        "currency": "EUR"
    },
    "description": "Order #12345",
    "method": null,
    "metadata": {
        "order_id": "12345"
    },
    "status": "open",
    "isCancelable": false,
    "expiresAt": "2018-03-20T09:28:37+00:00",
    "details": null,
    "profileId": "pfl_QkEhN94Ba",
    "sequenceType": "oneoff",
    "redirectUrl": "https://webshop.example.org/order/12345/",
    "webhookUrl": "https://webshop.example.org/payments/webhook/",
    "_links": {
        "self": {
            "href": "https://api.mollie.com/v2/payments/tr_7UhSN1zuX
            "type": "application/json"
        },
        "checkout": {
            "href": "https://www.mollie.com/payscreen/select-method/
            "type": "text/html"
        },
        "dashboard": {
            "href": "https://www.mollie.com/dashboard/org_12345678/p
            "type": "text/html"
        },
        "documentation": {
            "href": "https://docs.mollie.com/reference/v2/payments-a
            "type": "text/html"
        }
    }
}
```