



Adding the CB payment button

Implementation Guide

Document version 1.2.1

Contents

1. HISTORY OF THE DOCUMENT.....	4
2. PRESENTATION.....	5
2.1. Card types.....	6
Card categories.....	6
2.2. Authorization request.....	7
2.3. Information request.....	9
2.4. Recovery request.....	9
2.5. Choosing your favorite brand.....	10
2.6. 3D Secure.....	11
Merchant preference and Liability shift.....	11
Authentication scoring.....	12
2.7. Payment in foreign currency with conversion.....	13
2.8. Capture at the bank.....	13
2.9. Reconciling transactions and chargebacks.....	15
2.9.1. The “Visual reconciliation” service.....	15
2.9.2. The “Visual chargeback reconciliation” service.....	16
2.9.3. The “Reconciliation reports” service.....	16
2.9.4. The “Chargeback reconciliation report” service.....	16
2.10. Managing tokens and recurring payments.....	18
3. TECHNICAL INFORMATION.....	19
4. PREREQUISITES.....	20
5. INTEGRATION IN THE CUSTOMER JOURNEY.....	21
6. PAYMENT PROCESS.....	22
6.1. “Challenge” flow.....	22
6.2. “Frictionless” flow.....	23
7. ESTABLISHING INTERACTION WITH THE PAYMENT GATEWAY.....	24
8. SETTING UP NOTIFICATIONS.....	25
8.1. Setting up the Instant Payment Notification.....	26
8.2. Setting up the notification for the final result of a deferred payment.....	27
8.3. Setting up notifications in case of abandoned or canceled payments.....	28
9. GENERATING A PAYMENT FORM.....	29
9.1. Creating an immediate payment.....	31
9.2. Creating a deferred payment.....	33
9.3. Creating an installment payment.....	36
9.4. Creating a payment by token.....	39
9.5. Transmitting buyer details.....	40
9.6. Transmitting shipping details.....	41
9.7. Transmitting order details.....	42
9.8. Increasing the chances of a frictionless payment.....	44
9.9. Transmitting merchant preferences.....	46
10. SENDING THE PAYMENT REQUEST.....	47
10.1. Redirecting the buyer to the payment page.....	47
10.2. Processing errors.....	47
11. ANALYZING THE PAYMENT RESULT.....	48
11.1. Processing the response data.....	48

11.2. Analyzing the result of the authorization request.....	53
12. MANAGING CB TRANSACTIONS FROM THE MERCHANT BACK OFFICE.....	56
12.1. Viewing transaction details.....	56
12.2. Canceling a transaction.....	58
12.3. Duplicating a transaction.....	59
12.4. Modifying a transaction.....	61
12.5. Making a refund.....	62
12.6. Validating a transaction.....	63
12.7. Manual reconciliation.....	64
12.8. Capturing a transaction.....	65
13. OBTAINING HELP.....	66

1. HISTORY OF THE DOCUMENT

Version	Author	Date	Comment
1.2.1	La Banque Postale	1/23/2023	Update of chapters: <ul style="list-style-type: none">• <i>Integration in the customer journey</i>• <i>Payment in foreign currency with conversion</i>
1.2	La Banque Postale	12/22/2021	Update of the chapter <i>Processing the response data</i> .
1.1	La Banque Postale	8/25/2021	<i>Performing a refund</i> chapter: clarification on the refundable amount.
1.0	La Banque Postale	5/10/2021	Initial version

This document and its contents are confidential. It is not legally binding. Any reproduction and / or distribution of all or part of this document or its content to a third party is strictly prohibited or subject to prior written authorization from La Banque Postale. All rights reserved.

2. PRESENTATION

With more than 60% of current consumption being paid for with CB cards, CB has become the most widely used card and mobile payment scheme in France.

CB is an Economic Interest Group that defines the operating procedures for the CB card payment scheme (physical or electronic in case of mobile payments).

After being founded in 1984, CB has continued to develop card payments by incorporating the latest technological and security developments to make them ever more user-friendly, efficient and secure.

Payment by credit card is very simple and takes 2 steps:



1. Entering the CB card details (card number, expiry date and CVV)
2. 3D Secure cardholder authentication, required by the second Payment Services Directive (PSD2).

Card payment also allows to make:

- MOTO payment (Mail Order Telephone Order)
- Deferred payment
- Installment payment
- Recurring payment
- 1-click payment



Supported currencies

- Devises de votre domaine



Supported countries

- Worldwide*

**Contact us for more information*



Additional information

- The authorization request is valid for 7 days.
- Transaction capture is deferred (a capture delay set to 0 allows transactions to be captured as soon as possible).

2.1. Card types

A CB acceptance contract allows you to accept the following cards by default:

- CB
- e-Carte Bleue
- Maestro
- Mastercard
- Visa
- Visa Electron
- VPAY

For “restaurant owner” merchants who have declared their CB contract with the issuers of electronic Meal Vouchers, the CB contract allows them to accept:

- **1st generation Meal Voucher cards:** even if they have the logo of the Meal Voucher issuer, they have the distinction of being payment cards issued within the VISA or MASTERCARD networks. When it comes to authorization, they are processed similarly to Visa or Mastercard cards with a statutory daily limit.
- **Meal vouchers outside the Conecs network:** if you are affiliated with a Meal Voucher issuer other than Natixis Intertitres, Sodexo or Groupe Up, you can accept payments with these cards through your CB acceptance MID.

See the [Electronic Meal Vouchers](#) technical documentation available on our website.

Card categories

There are 4 categories of CB cards:

- **Credit card:** card whose amounts are debited on a deferred basis from the holder’s account, with or without interest.
- **Debit card:** payment card whose amounts are debited from the cardholder’s account less than 48 hours after the transactions were made. These are, for example, “immediate debit cards” or “cards with unconditional authorization”.
- **Commercial card:** used for business expenses and where the amounts are debited from the company account.
- **Prepaid card:** allows to have a limited amount of money. This card is exclusively reserved for private persons.

2.2. Authorization request

An authorization request is the operation that allows to accept or refuse a transaction.

It puts the cardholder's bank (SAE = Issuer Acceptance System) in contact with the merchant's bank (SAA = Acquirer Acceptance System) and the payment provider: the payment gateway.

When an authorization request is accepted, the authorization limit of the card is lowered by the authorized amount.

In the CB network, an accepted authorization request is valid:

- 7 days for Visa, Mastercard, Visa Electron, e-Carte Bleue and Vpay cards
- 30 days for Maestro cards

Deferred payment case

When the payment is deferred for more than 7 days, the payment gateway makes an **information request in real time** to check the card validity.

If the acquirer does not support information requests, the gateway proceeds to a EUR 1 authorization request.

The day before the capture date, the payment gateway makes an authorization request for the total amount.

If the request is accepted, the payment is captured at the bank on the scheduled date and the merchant can proceed with the delivery of the order.

Otherwise, the payment is refused.

Since the authorization request of the full amount is not made at the time of purchase, payments may be refused for insufficient balance.

To help merchants avoid losing these sales, the payment gateway offers an **anticipated authorizations service**.

This service allows the authorization to be triggered on **D-6** before the desired capture date (or D-29 for Maestro cards).

In case of refusal by the issuing bank, a process automatically reissues authorization requests until up to **2 days prior** the desired capture date at the bank.

This process only applies to certain refusal reasons (see table below).

The merchant may cancel the transaction or change its amount (only smaller amounts can be entered) and/or the capture date at any moment.

In case of refusal for a reason other than those listed in the table below, the transaction is considered definitively refused.

Below is a list of reasons that allow authorization retry:

Authorization return code	Description
00	Approved or successfully processed transaction

Authorization return code	Description
02	Contact the card issuer
08	Confirm after identification
17	Canceled by the buyer
19	Retry later
20	Incorrect response (error on the domain server)
24	Unsupported file update
25	Unable to locate the registered elements in the file
26	Duplicate registration, the previous record has been replaced
27	File update edit error
28	Denied access to file
29	Unable to update
30	Format error
38	Expired card
51	Insufficient balance or exceeded credit limit
55	Incorrect secret code
58	Transaction not allowed for this cardholder
60	The acceptor of the card must contact the acquirer
61	Withdrawal limit exceeded
68	Response not received or received too late
75	Number of attempts for entering the secret code has been exceeded
90	Temporary shutdown
91	Unable to reach the card issuer
94	Duplicate transaction
96	System malfunction
97	Overall monitoring timeout
98	Server not available, new network route requested
99	Initiator domain incident

Please contact the E-Banking Merchant Support Service if you wish to enable anticipated authorizations.

2.3. Information request

An information request is an operation that allows to verify the validity of the card, **without debiting it**.

This is a specific type of authorization request, the amount of which is 0€.

When the acquirer does not support information requests, the only way to verify a card is to make a EUR 1 authorization request, without capturing it at the bank.

Holders of prepaid and immediate debit cards will see a virtual debit of EUR 1 on their account.

Depending on the card type, the outstanding balance of the card is then restored when the issuer cancels the EUR 1 authorization request (up to 30 days for debit cards).

An information request is sent:

- For a deferred payment, if the capture date is beyond the authorization lifespan
- When creating a card token without a payment
- When updating a card token

Information requests (or EUR 1 authorizations, if applicable) are represented in the Merchant Back Office by a “**Verification**” type transaction.

2.4. Recovery request

A recovery request is the opposite of an authorization request.

It allows you to cancel an authorization request with the card issuer.

If the recovery request is accepted, the issuer should restore the card's outstanding balance, either in real time or on a delayed basis. Unfortunately this is not always the case, and in this case the cardholder will have their virtual debit canceled when the authorization expires.

To this date, all French buyers support this operation.

A recovery request is sent:

- when the merchant cancels a transaction, before it is captured at the bank,
- when an authorization request expires prior to merchant validation.

This is the case if the merchant has chosen to manually validate the payments before they are captured at the bank.

2.5. Choosing your favorite brand

The European (EU) **MIF (Merchant Interchange Fee) Regulation 215/751 of 29 April 2015** on interchange fees for card-related payment transactions is intended, among other purposes, to abolish the monopoly of the domestic brand: the consumer must be able to pay with his/her card on the Visa, Mastercard or CB network.

It allows:

- The merchant to choose the preferred brand that will be offered to buyers during the payment.
- The cardholder to have the choice of card brand (i.e. the network used for the transaction).

Thus, for CB cards co-branded VISA or MASTERCARD for example, the buyer will have the choice between the French CB network and the international VISA or MASTERCARD networks.

A co-branded card is a card that includes two payment companies.

Example

- **CB + VISA** or **ELECTRON** or **VPAY**
- **CB + MASTERCARD** or **MAESTRO**

Note

Each time an authorization uses international networks, the merchant may have to pay extra fees that will be transferred to the used network (depending on the commissioning methods of the merchant's acquirer contract).

What steps does the merchant need to take?

In Scellius, CB is configured as the default brand during the payment stages.

To change your preferred brand, contact your bank to request the update.

What impact on the payment process?

While the buyer enters the card number, the payment gateway searches for brands associated with the card.

The search is based on the BIN (Bank Identification Number) of the card. Once the list of brands has been identified, there are several possibilities:

- **The card is associated with only one brand**

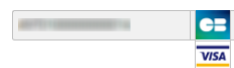
In this case, the logo of the corresponding brand is automatically displayed.



- **The card is associated with several brands**

In this case, the logo of the merchant's preferred brand is automatically displayed.

A drop-down list will appear to the right of the entry field to allow the buyer to choose another brand.



- **No brands detected**

The buyer possibly made an error when entering the card number.



For more information, see our [Brand selection](#) guide.

2.6. 3D Secure

3D Secure is an interbank protocol that provides a high level of security for online payments.

In 2019, CB developed its own service for securing card payments called “FAST'R by CB”.

It acts as a Directory Server for strong cardholder authentication, but also as an anti-fraud tool thanks to an authentication scoring system and merchant risk management.

When a transaction is processed by the CB network, the “**CB Paiement sécurisé**” logo reassures the buyer that the payment is secure and that it is processed in France.



The second Payment Services Directive (or PSD2) requires strong authentication for payments when the buyer is present at the time of purchase, but also provides for cases where interaction with the buyer (challenge) is not mandatory. To qualify for frictionless authentication, the payment must be eligible for an exemption.

Merchant preference and Liability shift

Under PSD2, it is no longer possible to disable authentication in 3DS2.

However, the merchant can express their choice regarding cardholder authentication.

This is called “**merchant preference**”.

The merchant can choose to:

- Request strong authentication, i.e. with cardholder interaction (challenge)
- Request authentication without interaction (frictionless)
- Not choose anything and let the issuer decide (no preference)

By default, “no preference” is applied.

The choice is made either in the payment request, or via a payment module (PrestaShop, Magento, etc.), or via the Merchant Back Office for merchants authorized to access the advanced risk module.

The expression of this wish is taken into account in the CB scoring and is communicated to the issuer. In addition, this desire has an impact on the transfer of responsibility.

	If issuer applies passive authentication	If issuer applies strong authentication
Merchant preference "passive authentication"	Cost of fraud liable by the merchant	Cost of fraud liable by the issuer
Merchant preference "strong authentication"	Cost of fraud liable by the issuer	Cost of fraud liable by the issuer
Merchant preference "no preference"	Cost of fraud liable by the issuer	Cost of fraud liable by the issuer

The burden of fraud is always borne by the issuer, except in the case where the merchant requests passive authentication (frictionless) and the issuer enforces this choice.

For more information, please see our [3D Secure](#) guide.

Authentication scoring

For each authentication request on the CB network, a score is calculated.

This score reflects the level of risk and is based on the current transaction data as well as the historical purchase profile of the cardholder and the merchant.

The score is between 0 and 99.

- It is sent to the ACS in the authentication request, to facilitate the issuer's decision (strong or frictionless authentication).
- It is sent in the return of the authentication request to share the risk analysis with all the actors of the payment chain.
- It is sent by the payment platform in the authorization request.

Based on the score achieved, CB will provide the issuer with a recommendation that depends on other factors, including the merchant's desire for strong authentication.

FAST'R by CB scoring value	1	2	...	30	31	32	33	...	97	98	99
Fraud risk	0,01%	0,02%		0,30%	0,31 - 1%	2%	3%		67%	68 - 98%	99%

2.7. Payment in foreign currency with conversion

Payment in foreign currency with conversion allows Merchants to present price catalogs in different currencies, but without having to manage their finances in currencies different from the ones specified in their contract.

When the gateway receives the amount in a currency not managed by your MIDs, it makes a conversion to the company's currency based on the daily exchange rate provided by Visa.

The buyer is informed of the indicative rate at the time of payment, but does not really know the final amount of the transaction.

The capture at the bank does not necessarily occur on the day of the authorization and the rate may therefore vary between the date of authorization and the date of capture.

For this reason, the counter-value displayed at the time of payment is provided as an indication.

Important:



- The authorization request is sent in the currency of the contract to the card issuer.
- The capture is performed exclusively in the currency of the contract.
- The buyer is debited in the contract currency with exchange fees applied by their bank, without managing the exchange rate.

At the end of the payment, the merchant receives a notification containing the following fields:

- **vads_amount**: the currency amount,
- **vads_currency**: the currency,
- **Vads_effective_amount**: the actual amount in the currency of their contract, calculated using the exchange rate in force at the time of the authorization,
- **vads_effective_currency**: the currency that will be used for the capture,
- **vads_change_rate**: the exchange rate applied for converting the amount in the currency of the contract to the buyer's currency.

2.8. Capture at the bank

Once a day, the payment gateway sends the acquirer files containing the debit and credit transactions (refunds) to be captured at the bank.

Card verification transactions are not captured at the bank.

Depending on the volume of transactions to be remitted, the payment gateway can generate multiple captures for a single capture date.

The capture files are sent in the evening, between 10:30 pm and 2 am Paris time.

The acquirer processes the capture files (pre-clearing phase) and then sends a clearing file to each issuer in order to finalize the transaction.

The amounts are then paid into the accounts of the various payment actors:

- From the buyer to the merchant for debit transactions

- From the merchant to the buyer for credit transactions (refunds)

Processing times may vary depending on the acquirer or card issuer.

The merchant can view the capture details in their Merchant Back Office (**Management** > **Transactions** menu, **Captures** tab).

2.9. Reconciling transactions and chargebacks

CB transactions are supported by the following services:

- Visual transaction reconciliation,
- Visual chargeback reconciliation,
- Bank reconciliation report,
- Chargeback reconciliation report.

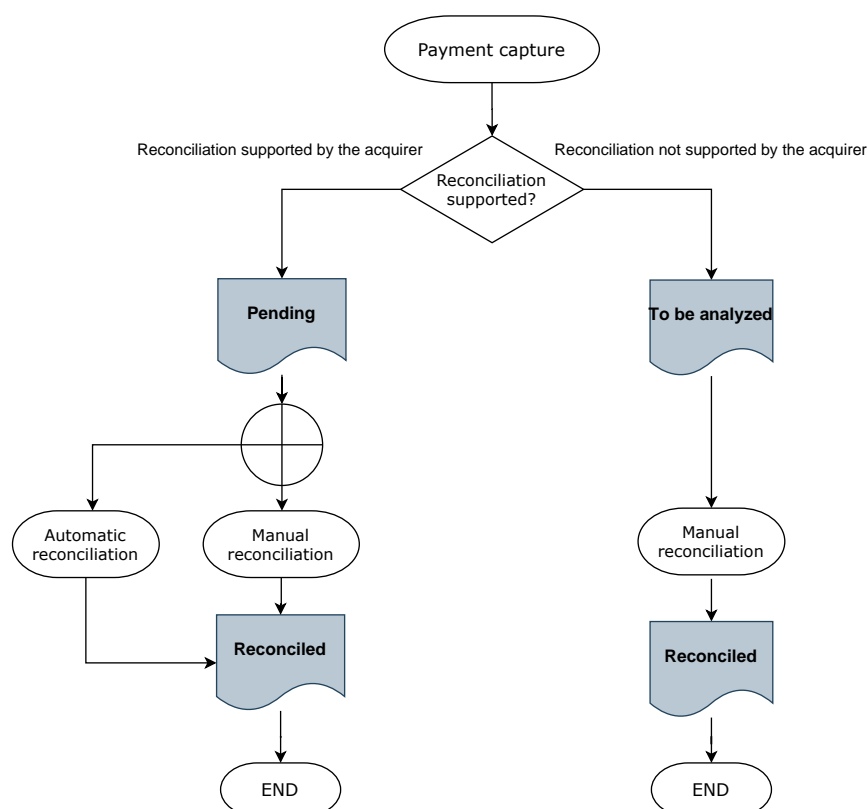
If you want to enable these services:

1. Request the activation of reconciliation flows from your bank,
2. Contact the E-Banking Merchant Support Service to enable transaction reconciliation via the payment gateway.

2.9.1. The “Visual reconciliation” service

The visual reconciliation service allows merchants to benefit from automatic reconciliation of transactions made on the payment gateway with the payments that appear on their bank statement.

Operating principle



If the acquirer supports reconciliation, each captured transaction gets the “**Pending**” reconciliation status.

If the acquirer does not support reconciliation, captured transactions get the “**To be analyzed**” reconciliation status.

As soon as the service is enabled, automatic processing is set up to check the transactions acquired on the payment gateway and the bank statement entries.

Successfully reconciled transactions get the **“Reconciled”** status.

Transactions with the **“Pending”** or **“To be analyzed”** status may be reconciled manually by the merchant via the **Captured transactions** tab (see chapter [Manual reconciliation](#) on page 64).

2.9.2. The “Visual chargeback reconciliation” service

The visual chargeback reconciliation service allows merchants to benefit from automatic reconciliation of chargeback transactions.

Operating principle

Each captured transaction gets a **“Dispute”** chargeback reconciliation status set to:

- **Yes:** the visual chargeback reconciliation service is enabled and a dispute has been filed for the transaction regardless of the dispute outcome,
- **No:** the visual chargeback reconciliation service is enabled and no disputes have been filed for the transaction
- **N/A:** the visual chargeback reconciliation service is disabled.

As soon as the service is enabled, automatic processing is set up to check the transactions acquired on the payment gateway and any potential chargebacks.

Transactions that are subject to chargeback reconciliation get the **“Yes”** status for the **“Dispute”** piece of data.

Transactions that are not subject to chargeback reconciliation, while the option is selected, get the **“No”** status.

The **“N/A”** status corresponds to transactions of a merchant that has not opted for the visual chargeback reconciliation service.

List of acquirers supporting chargeback reconciliation:

Network CODE	Acquirer
AMEXGLOBAL	American Express Global
CB	La Banque Postale

2.9.3. The “Reconciliation reports” service

This service provides the merchant with a file containing a list of automatically reconciled transactions. The report provides additional bank details (amount credited to the account, commission fees, etc.).

CB transactions are identified by the **CB** value of the **CARD_TYPE** (V1 format) or **PAYMENT_METHOD** (V3 format) data contained in the report.

For more information on this service, please see the [Description of reporting](#) guide available in our online document archive.

2.9.4. The “Chargeback reconciliation report” service

This service provides the merchant with a file containing a list of automatically reconciled chargeback transactions. The report provides additional bank details (reason of the chargeback, etc.).

CB transactions are identified by the **CB** value of the **CARD_TYPE** data item contained in the report.

For more information on this service, please see the [Description of reporting](#) guide available in our online document archive.

2.10. Managing tokens and recurring payments

CB transactions are supported by the following services:

- [*Management of payments by token*](#)
- [*Recurring payment \(subscription\) management*](#)

For more information on these services, see the corresponding documentation available in our online document archive.

If you wish to enable or obtain more information on these services, contact the E-Banking Merchant Support Service.

3. TECHNICAL INFORMATION

Values of the vads_card_brand field	Supported currencies	Supported countries	Authorization validity period	Payment process
CB VISA MASTERCARD MAESTRO VISA_ELECTRON E-CARTEBLEUE VPAY APETIZ* CHQ_DEJ* SODEXO* * Only for merchants registered with a restaurant owner MCC.	EUR Possibility to perform the purchase in another currency if the acquirer supports it. Possibility to accept payments in a currency that it different from the one specified in the contract, with conversion to euro.	No restriction	7 days 30 days for Maestro	Deferred capture

Sales channel	
e-commerce	✓
m-commerce	✓
MOTO payment	✓
Payment order by e-mail/SMS	✓

Operations with transactions	
Cancellation	✓
Refund	✓
Modification	✓
Validation	✓
Duplication	✓
Manual reconciliation	✓

Type of integration	
Redirection	✓
Iframe	✓
JavaScript Client	✓
Data collection form	✓
API Web Services	✓
Mobile SDK	✓
Back Office	✓

Payment type	
Immediate payment	✓
Deferred payment	✓
Payment in installments	✓
Payment by subscription	✓
Payment by file (token or batch)	✓
One-click payment	✓
Payment by wallet	✓

Miscellaneous	
Reporting	✓
Transaction settlement	✓
Chargeback settlement	✓
Extra payment attempts	✓

4. PREREQUISITES

To accept CB payments on your e-commerce site, you must ask your bank to open a VADS (Secure Distance Selling) acceptance contract.

Once you have your contract number, contact the E-Banking Merchant Support Service to declare your CB contract number.




5. INTEGRATION IN THE CUSTOMER JOURNEY


In order to simplify the customer journey, increase the conversion rate and thus reduce the number of abandoned orders, it is recommended to:


- select the payment method on the merchant website,
- generate a payment button for each type of payment method.

Example of payment method selection:

Select a payment method:









A unique logo regrouping CB, VISA and MASTERCARD allows to arrive directly on the data page of the card seizures if the merchant has only one CB contract. If the merchant has several payment methods, then CB, VISA, MASTERCARD, E-CARTE BLEUE, MAESTRO are grouped under a single logo on the payment page.

6. PAYMENT PROCESS

6.1. “Challenge” flow

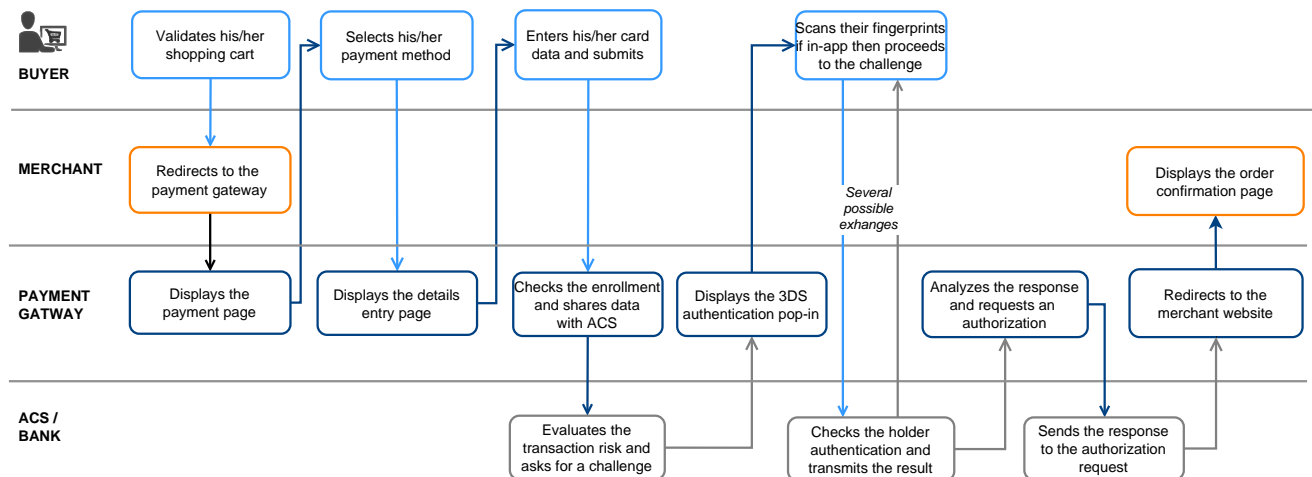
In a challenge flow, based on the received information, the issuer determines that it is necessary for the buyer to provide the following elements:

- Either a biometric element, such as a device fingerprint,
- or a strong authentication via two-factor authentication.

For in-app solutions, the device fingerprint will be systematically requested before proceeding to the challenge.

Once the challenge has been successfully completed, the payment gateway proceeds with the payment and issues the authorization request.

In case of a technical or authentication error, the payment stops. The payment gateway notifies the merchant website and the buyer about the payment rejection and redirects the buyer to the merchant website.



6.2. “Frictionless” flow

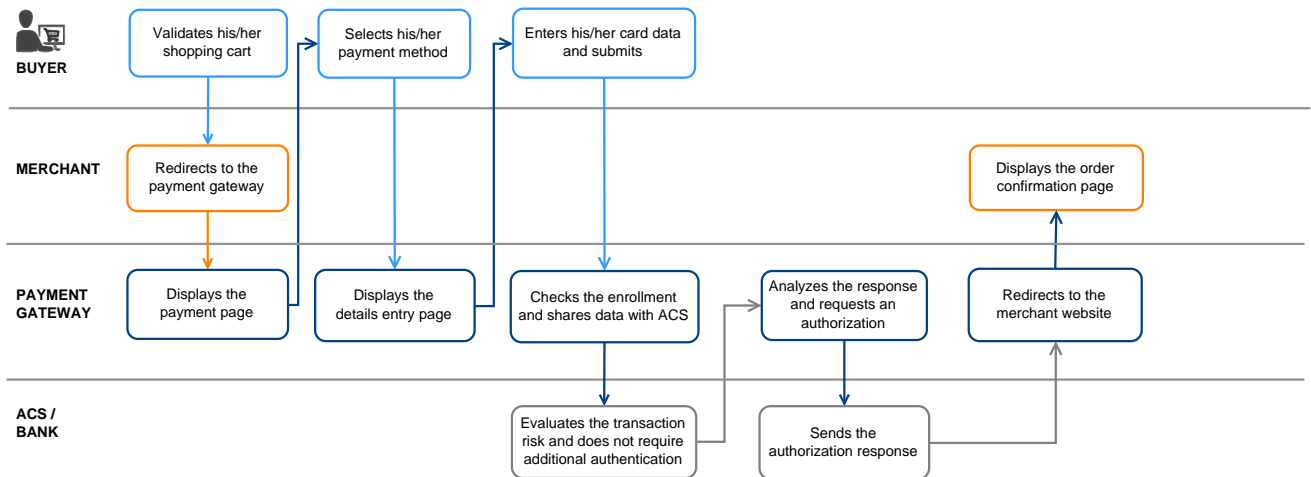
In frictionless flow (without interaction with the buyer), based on the received information, the issuer can determine:

- That no additional authentication is required.

The payment gateway proceeds with the payment and issues the authorization request.

- That the analyzed information does not provide the authorization to proceed with the payment.

In this case, the payment gateway notifies the merchant website and the buyer about the payment rejection and redirects the buyer to the merchant website.



7. ESTABLISHING INTERACTION WITH THE PAYMENT GATEWAY

The interaction with the payment gateway is described in the *Implementation Guide Hosted Payment Page*, available in our online documentation archive.

The merchant website and the payment gateway interact by exchanging data.

To create a payment, this data is sent in an HTML form via the buyer's browser.

At the end of the payment, the result is transmitted to the merchant website in two ways:

- automatically by means of notifications called Instant Notification URLs (also known as IPN or Instant Payment Notification).
- Via the browser when the buyer clicks the button to return to the merchant website.

To guarantee the security of the exchange, the data is signed with a key known only to the merchant and the payment gateway.

8. SETTING UP NOTIFICATIONS

The Merchant Back Office provides several types of notifications.

- Instant Payment Notification URL call
- E-mail sent to the merchant
- E-mail sent to the buyer
- SMS sent to the merchant
- SMS sent to the buyer

They allow to manage the events (payment accepted, payment abandoned by the buyer, payment canceled by the merchant, etc.) that will trigger a notification sent to the merchant website, the merchant or the buyer.



The notifications of Instant Payment Notification URL call type are very important as they represent the only reliable way for the merchant website to obtain the payment result.

If the payment gateway is unable to access the URL of your page, an e-mail will be sent to the shop administrator.

It contains:

- The HTTP code of the encountered error
- Parts of error analysis
- Its consequences
- Instructions via the Merchant Back Office to resend the request to the previously defined URL.

To access notification rule management:

Go to the following menu **Settings > Notification rules**.

Instant Payment Notification		E-mail sent to the merchant	E-mail sent to the buyer
Enabled			
✗	Instant Payment Notification URL on batch authorization		
✓	Instant Payment Notification URL at the end of the payment		
✗	Instant Payment Notification URL on batch change		
✗	Instant Payment Notification URL on cancellation		
✗	Instant Payment Notification URL on an operation coming from the Back Office		

8.1. Setting up the Instant Payment Notification

This rule allows to notify the merchant website in the following cases:

- Payment accepted
- Payment refused
- Token creation or update
- Creation of a recurring payment



This notification is required for communicating the result of a payment request, token or recurring payment creation.

It will inform the merchant website of the result even if the buyer has not clicked the “Return to the shop” button.

1. Right-click **Instant Payment Notification URL at the end of the payment**.
2. Select **Manage the rule**.
3. Enter the **E-mail address(es) to notify in case of failure** field in the **General settings** section.
To specify several e-mail addresses, separate them with a semi-colon.
4. Check the box **Automatic retry in case of failure** if you wish to authorize the gateway to automatically resend the notification in case of a failure (can be done up to 4 times).
5. In the **Instant Payment Notification URL of the API form V1, V2** section, specify the URL of your page in the fields **URL to notify in TEST mode** and **URL to notify in PRODUCTION mode** if you wish to receive notifications in the API form format.
6. Save the changes.

8.2. Setting up the notification for the final result of a deferred payment

This notification is required for communicating the result of a deferred payment:

- If the payment has been accepted,
- If the payment has been refused.

It allows the merchant website to be notified when the authorization request is not made on the payment day.

Example:

For a deferred payment with a capture delay of 60 days, the authorization request is not made at the moment of the payment. The merchant website will be contacted at the moment of the authorization request by the **Instant Payment Notification URL on batch authorization** rule.

This rule is **disabled by default**.

1. Right-click **Instant Payment Notification URL on batch authorization**.
2. Select **Manage the rule**.
3. Enter the **E-mail address(es) to notify in case of failure** field in the **General settings** section.
To specify several e-mail addresses, separate them with a semi-colon.
4. Check the box **Automatic retry in case of failure** if you wish to authorize the gateway to automatically resend the notification in case of a failure (can be done up to 4 times).
5. In the **Instant Payment Notification URL of the API form V1, V2** section, specify the URL of your page in the fields **URL to notify in TEST mode** and **URL to notify in PRODUCTION mode** if you wish to receive notifications in the API form format.
6. In the **REST API Instant Payment Notification URL** section, specify the URL of your page in the fields **Target URL of the IPN to notify in TEST mode** and **Target URL of the IPN to notify in PRODUCTION mode** if you are using the JavaScript client.
7. Save the changes.
8. Enable the rule by right-clicking **Instant Payment Notification URL on batch authorization** and select **Enable the rule**.

8.3. Setting up notifications in case of abandoned or canceled payments

This rule allows to notify the merchant website in the following cases:

- When the buyer abandons/cancels a payment - via the **Cancel and return to shop** button.
- When the buyer has not completed the payment and the payment session has expired.

The maximum length of a payment session is 10 minutes.

This rule is **disabled by default**.

1. Right-click **Instant Payment Notification URL on cancellation**.
2. Select **Manage the rule**.
3. Enter the **E-mail address(es) to notify in case of failure** field in the **General settings** section.
To specify several e-mail addresses, separate them with a semi-colon.
4. Check the box **Automatic retry in case of failure** if you wish to authorize the gateway to automatically resend the notification in case of a failure (can be done up to 4 times).
5. In the **Instant Payment Notification URL of the API form V1, V2** section, specify the URL of your page in the fields **URL to notify in TEST mode** and **URL to notify in PRODUCTION mode** if you wish to receive notifications in the API form format.
6. Save the changes.
7. Enable the rule by right-clicking **Instant Payment Notification URL on cancellation** and select **Enable the rule**.

9. GENERATING A PAYMENT FORM

To generate a payment request, you must create an HTML form as follows:

```
<form method="POST" action="https://scelliuspaiement.labanquepostale.fr/vads-payment/">
  <input type="hidden" name="parameter1" value="value1" />
  <input type="hidden" name="parameter2" value="value2" />
  <input type="hidden" name="parameter3" value="value3" />
  <input type="hidden" name="signature" value="signature"/>
  <input type="submit" name="pay" value="Pay"/>
</form>
```

It contains:

- The following technical elements:
 - The `<form>` and `</form>` tags that allow to create an HTML form.
 - The `method="POST"` attribute that defines the method used for sending data.
 - The `action="https://scelliuspaiement.labanquepostale.fr/vads-payment/"` attribute that defines where to send the form data.
- Form data:
 - The shop ID.
 - Information about the payment depending on the use case.
 - Additional information depending on your needs.
 - The signature that ensures the integrity of the form.

This data is added to the form by using the `<input>` tag:

```
<input type="hidden" name="parameter1" value="value1" />
```

For setting the `name` and `value` attributes, see the **Data dictionary** chapter also available in the online document archive.

All the data in the form must be encoded in **UTF-8**.

This will allow for the special characters (accents, punctuation marks, etc.) to be correctly interpreted by the payment gateway. Otherwise, the signature will be computed incorrectly and the form will be rejected.

- The **Pay** button for submitting the data:

```
<input type="submit" name="pay" value="Pay"/>
```

Different use cases are presented in the chapters below. They will allow you to adapt your payment form to your needs.

The following table lists the different formats that you can encounter when building your form.

Notation	Description
a	Alphabetic characters (from 'A' to 'Z' and from 'a' to 'z')
n	Numeric characters
s	Special characters
an	Alphanumeric characters
ans	Alphanumeric and special characters (except '<' and '>')
3	Fixed length of 3 characters
..12	Variable length up to 12 characters
json	<p>JavaScript Object Notation.</p> <p>Object containing key/value pairs separated by commas.</p> <p>It starts with a left brace "{" and ends with a right brace "}".</p> <p>Each key / value pair contains the name of the key between double-quotes followed by ":", followed by a value.</p> <p>The name of the key must be alphanumeric.</p> <p>The value can be:</p> <ul style="list-style-type: none"> • a chain of characters (in this case it must be framed by double-quotes) • a number • an object • a table • a boolean • empty <p>Example: {"name1":45,"name2":"value2","name3":false}</p>
bool	Boolean. Can be populated with the true or false value.
enum	<p>Defines a field with a complete list of values.</p> <p>The list of possible values is given in the field definition.</p>
Enum list	<p>List of values separated by a ";".</p> <p>The list of possible values is given in the field definition.</p> <p>Example: vads_available_languages=fr;en</p>
map	<p>List of key / value pairs separated by a ";".</p> <p>Each key / value pair contains the name of the key followed by "=", followed by a value.</p> <p>The value can be:</p> <ul style="list-style-type: none"> • a chain of characters • a boolean • a json object • an xml object <p>The list of possible values for each key/value pair is provided in the field definition.</p> <p>Example: vads_theme_config=SIMPLIFIED_DISPLAY=true;RESPONSIVE_MODEL=Model_1</p>

9.1. Creating an immediate payment

A payment is considered as **immediate payment** if:

- the amount is debited once,
- the capture delay at the bank is 0 days.

An authorization request for the total amount is sent. The payment is captured at the bank as soon as possible.

1. Use all the fields presented in the table below to create your payment form.

Field name	Description	Format	Value
vads_payment_cards	Allows to force the card type to be used. It is recommended to provide a different payment button for each payment method on the merchant website. It is recommended not to leave the field empty.	enum	AMEX to directly call the CB payment button.
vads_site_id	Shop ID	n8	E.g.: 12345678
vads_ctx_mode	Mode of interaction with the payment gateway	enum	TEST or PRODUCTION
vads_trans_id	Transaction number. Must be unique within the same day (from 00:00:00 UTC to 23:59:59 UTC). Warning: this field is not case sensitive.	an6	E.g.: xrT15p
vads_trans_date	Date and time of the payment form in UTC format	n14	Respect the YYYYMMDDHHMMSS format E.g.: 20200101130025
vads_amount	Payment amount in the smallest currency unit (cents for euro)	n..12	E.g.: 4525 for EUR 45.25
vads_currency	Numeric currency code to be used for the payment, in compliance with the ISO 4217 standard (numeric code).	n3	E.g.: 978 for euro (EUR)
vads_action_mode	Acquisition mode for payment method data	enum	INTERACTIVE
vads_page_action	Action to perform	enum	PAYMENT
vads_version	Version of the exchange protocol with the payment gateway	enum	V2
vads_payment_config	Payment type	enum	SINGLE
vads_capture_delay	Capture delay	n..3	0
vads_validation_mode	Validation mode	n1	0 (Automatic)

2. Set the **vads_payment_config** field to **SINGLE**.
3. Set the **vads_capture_delay** field to **0**.
4. Set the **vads_validation_mode** field to **0** for automatic validation (the payment will be automatically captured in the bank).
5. Add *the fields recommended for increasing chances of frictionless* during the payment.
6. Add optional fields according to your requirements (see **Using additional features** chapter of the *Implementation Guide Hosted Payment Page*).
7. Compute the value of the **signature** field using all the fields of your form that start with **vads_** (see chapter **Computing the signature** of the *Implementation Guide Hosted Payment Page* available on our website).

Example of a form for an immediate payment:

```
<form method="POST" action="https://scelliuspaiement.labanquepostale.fr/vads-payment/">
<input type="hidden" name="vads_action_mode" value="INTERACTIVE" />
<input type="hidden" name="vads_amount" value="15000" />
<input type="hidden" name="vads_capture_delay" value="0" />
<input type="hidden" name="vads_ctx_mode" value="TEST" />
<input type="hidden" name="vads_currency" value="978" />
<input type="hidden" name="vads_order_id" value="CX-1254" />
<input type="hidden" name="vads_page_action" value="PAYMENT" />
<input type="hidden" name="vads_payment_cards" value="AMEX" />
<input type="hidden" name="vads_payment_config" value="SINGLE" />
<input type="hidden" name="vads_site_id" value="12345678" />
<input type="hidden" name="vads_trans_date" value="20190626101407" />
<input type="hidden" name="vads_trans_id" value="pt156G" />
<input type="hidden" name="vads_version" value="V2" />
<input type="hidden" name="signature" value="0WaYrONo3L0VZqMcvyVf8vT/g8KfZKJ+1jqAs3Ehiw=" />
<input type="submit" name="payer" value="Payer"/>
</form>
```


9.2. Creating a deferred payment

A payment is considered a **deferred payment** if:

- the amount is debited once,
- the capture delay is strictly more than 0 days.

The capture date cannot be more than 12 months after the payment request registration date.

There are two types of deferred payments:

- **Capture delay less than 7 days (or less than 30 days for a Maestro card)**

An authorization request for the total amount is sent. If the merchant has not made any modifications, the payment is captured by the bank on the requested capture day.

- **Capture delay more than 7 days (or more than 30 days for a Maestro card)**

An information request will be made if the capture delay is greater than the validity period of an authorization request.

The information request is made in order to check the card validity. For acquirers who do not support information requests, an authorization request for EUR 1 will be made.

If this authorization for EUR 1 is accepted, the payment request is registered.

An authorization request for the total amount is made one day before the requested capture.

The result is notified to the merchant website due to the [Instant Payment Notification URL on batch authorization](#) rule.

The payment might be accepted or refused. Therefore, the merchant must be extremely attentive with this payment type and make sure to deliver the items/services to the buyer.

1. Use all the fields presented in the table below to create your payment form.

Field name	Description	Format	Value
vads_payment_cards	Allows to force the card type to be used. It is recommended to provide a different payment button for each payment method on the merchant website. It is recommended not to leave the field empty.	enum	AMEX to directly call the CB payment button.
vads_site_id	Shop ID	n8	E.g.: 12345678
vads_ctx_mode	Mode of interaction with the payment gateway	enum	TEST or PRODUCTION
vads_trans_id	Transaction number. Must be unique within the same day (from 00:00:00 UTC to 23:59:59 UTC). Warning: this field is not case sensitive.	an6	E.g.: xrT15p
vads_trans_date	Date and time of the payment form in UTC format	n14	Respect the YYYYMMDDHHMMSS format E.g.: 20200101130025

Field name	Description	Format	Value
vads_amount	Payment amount in the smallest currency unit (cents for euro)	n..12	E.g.: 4525 for EUR 45.25
vads_currency	Numeric currency code to be used for the payment, in compliance with the ISO 4217 standard (numeric code).	n3	E.g.: 978 for euro (EUR)
vads_action_mode	Acquisition mode for payment method data	enum	INTERACTIVE
vads_page_action	Action to perform	enum	PAYMENT
vads_version	Version of the exchange protocol with the payment gateway	enum	V2
vads_payment_config	Payment type	enum	SINGLE
vads_capture_delay	Delay before capture in the bank, the value must be greater than 0	n..3	E.g.: 3
vads_validation_mode	Specifies the validation mode of the transaction (manually by the merchant, or automatically by the payment gateway).	n1	0 or 1 or absent or empty

2. Set the **vads_payment_config** field to **SINGLE**.
3. Set the **vads_capture_delay** field to a value **greater than 0**.
4. Set the **vads_validation_mode** field to **0** for an automatic validation (the payment will be automatically captured at the bank) or to **1** for a manual validation (the payment will be captured in the bank after a manual validation in the Merchant Back Office).
5. Add *the fields recommended for increasing chances of frictionless* during the payment.
6. Add optional fields according to your requirements (see **Using additional features** chapter of the *Implementation Guide Hosted Payment Page*).
7. Compute the value of the **signature** field using all the fields of your form that start with **vads_** (see chapter **Computing the signature** of the *Implementation Guide Hosted Payment Page* available on our website).

Example of a form for a deferred payment:

```
<form method="POST" action="https://scelliuspaiement.labanquepostale.fr/vads-payment/">
<input type="hidden" name="vads_action_mode" value="INTERACTIVE" />
<input type="hidden" name="vads_amount" value="3000" />
<input type="hidden" name="vads_capture_delay" value="3" />
<input type="hidden" name="vads_ctx_mode" value="TEST" />
<input type="hidden" name="vads_currency" value="978" />
<input type="hidden" name="vads_page_action" value="PAYMENT" />
<input type="hidden" name="vads_payment_cards" value="CB" />
<input type="hidden" name="vads_payment_config" value="SINGLE" />
<input type="hidden" name="vads_site_id" value="12345678" />
<input type="hidden" name="vads_trans_date" value="20190629130025" />
<input type="hidden" name="vads_trans_id" value="Hu92ZQ" />
<input type="hidden" name="vads_version" value="V2" />
<input type="hidden" name="signature" value="NrHSHyBBBc+TtcaudspNHQ5cYcy4tS4IjvdC0ztFe8=" />
<input type="submit" name="pay" value="Pay"/>
```

</form>

9.3. Creating an installment payment



Under PSD2, strong authentication is required upon the payment of the first installment. The `vads_threeds_mpi` field is ignored and an authentication with buyer interaction is automatically requested.

This payment mode allows the merchant to offer payment facilities to the buyer.

The payment form defines the number of installments and the interval between them.

The schedule is then presented to the buyer on the payment pages.

The first installment works the same way as an immediate payment. The result is notified to the merchant website due to the [*Instant Payment Notification URL at the end of the payment*](#) rule.

The following maturities are similar to deferred cash payments. The result is notified to the merchant website due to the [*Instant Payment Notification URL on batch authorization*](#) rule.

Clarifications:

The `vads_amount` field contains the total amount of the order. This is the amount that will be split according to the value of the `vads_payment_config` field.

On the day of the payment, the total amount is not credited to the merchant's account and the liability shift cannot apply to future installments.

The date of the last installment cannot exceed one year after the date of the form submission. Otherwise, an error message will appear and the form will be rejected.

1. Use all the fields below to create your payment form.

Field name	Description	Format	Value
<code>vads_payment_cards</code>	Allows to force the card type to be used. It is recommended to provide a different payment button for each payment method on the merchant website. It is recommended not to leave the field empty.	enum	AMEX to directly call the CB payment button.
<code>vads_site_id</code>	Shop ID	n8	E.g.: 12345678
<code>vads_ctx_mode</code>	Mode of interaction with the payment gateway	enum	TEST or PRODUCTION
<code>vads_trans_id</code>	Transaction number. Must be unique within the same day (from 00:00:00 UTC to 23:59:59 UTC). Warning: this field is not case sensitive.	an6	E.g.: xrT15p
<code>vads_trans_date</code>	Date and time of the payment form in UTC format	n14	Respect the YYYYMMDDHHMMSS format E.g.: 20200101130025

Field name	Description	Format	Value
vads_amount	Payment amount in the smallest currency unit (cents for euro)	n..12	E.g.: 4525 for EUR 45.25
vads_currency	Numeric currency code to be used for the payment, in compliance with the ISO 4217 standard (numeric code).	n3	E.g.: 978 for euro (EUR)
vads_action_mode	Acquisition mode for payment method data	enum	INTERACTIVE
vads_page_action	Action to perform	enum	PAYMENT
vads_version	Version of the exchange protocol with the payment gateway	enum	V2
vads_payment_config	Payment type	enum	See step 2.
vads_capture_delay	Capture delay	n..3	0
vads_validation_mode	Specifies the validation mode of the transaction (manually by the merchant, or automatically by the payment gateway).	n1	0 or 1 or absent or empty

2. Populate the **vads_payment_config** field using the following syntax:

- Fixed payment amounts and dates:

MULTI:first=1000;count=3;period=30 where:

"first" corresponds to the amount (in the smallest currency unit) of the first installment made on the day of payment,

"count" represents the total number of installments,

"period" determines the interval between each installment.

- Custom installment amounts and dates:

MULTI_EXT:date1=amount1;date2=amount2;date3=amount3 where:

date1=amount1 defines the date and the amount of the first transfer.

The amounts are presented in the smallest currency unit. The total amount must be equal to the value of the **vads_amount** field.

The dates are presented in the YYYYMMDD format.

3. Set the **vads_capture_delay** field to **0**. The first payment will be captured in the bank on the same day.
4. Set the **vads_validation_mode** field to **0** for automatic validation (the payment will be automatically captured in the bank) or to **1** for manual validation (manual operation performed via the Merchant Back Office).
The validation mode applies to all the installments.
5. Add optional fields according to your requirements (see **Using additional features** chapter of the [Implementation Guide Hosted Payment Page](#)).

6. Compute the value of the **signature** field using all the fields of your form that start with **vads_** (see chapter **Computing the signature** of the [Implementation Guide Hosted Payment Page](#) available on our website).

Example of installment payment form (fixed amounts and payment dates):

```
<form method="POST" action="https://scelliuspaiement.labanquepostale.fr/vads-payment/">
<input type="hidden" name="vads_action_mode" value="INTERACTIVE" />
<input type="hidden" name="vads_amount" value="3000" />
<input type="hidden" name="vads_ctx_mode" value="TEST" />
<input type="hidden" name="vads_currency" value="978" />
<input type="hidden" name="vads_page_action" value="PAYMENT" />
<input type="hidden" name="vads_payment_config" value="MULTI:first=1000;count=3;period=30"/>
<input type="hidden" name="vads_site_id" value="12345678" />
<input type="hidden" name="vads_trans_date" value="20190629180150" />
<input type="hidden" name="vads_trans_id" value="1N015m" />
<input type="hidden" name="vads_version" value="V2" />
<input type="hidden" name="signature" value="zrhUNkAcizSEl6mS4BbhV3qkYUBB9RYJQCdglkU0ELU=" />
<input type="submit" name="pay" value="Pay" />
</form>
```

Example of installment payment form (custom amounts and payment dates):

```
<form method="POST" action="https://scelliuspaiement.labanquepostale.fr/vads-payment/">
<input type="hidden" name="vads_action_mode" value="INTERACTIVE" />
<input type="hidden" name="vads_amount" value="3000" />
<input type="hidden" name="vads_capture_delay" value="0" />
<input type="hidden" name="vads_ctx_mode" value="TEST" />
<input type="hidden" name="vads_currency" value="" />
<input type="hidden" name="vads_page_action" value="PAYMENT" />
<input type="hidden" name="vads_payment_config" value="
MULTI_EXT:20140201=1000;20140301=1000;20140401=1000" />
<input type="hidden" name="vads_site_id" value="12345678" />
<input type="hidden" name="vads_trans_date" value="20190629130025" />
<input type="hidden" name="vads_trans_id" value="130025" />
<input type="hidden" name="vads_version" value="V2" />
<input type="hidden" name="signature" value="7Sds6Z+RlQ1axRsblpChyQh5OU3oCle5FOirD4V/Bzk=" />
<input type="submit" name="pay" value="Pay" />
</form>
```

9.4. Creating a payment by token

Payment by token (or one-click payment) allows the buyer to pay without entering their card data. For this, the buyer must have accepted the registration of his card by the payment gateway.

For more information about registering a card, please see [Payment by token - Recurring payment](#) guide available on our website.



In the context of the application of the PSD2, the CVV entry as well as cardholder authentication are required during a payment by token, as long as the buyer is present. The `vads_threeds_mpi` field is taken into account to allow for potential authentication without buyer interaction.

1. Use all the fields presented in the table below to create your form.

Field name	Description	Format	Value
vads_page_action	Action to perform	enum	PAYMENT
vads_amount	Payment amount (in the smallest currency unit)	n..12	E.g.: 4525 for EUR 45.25
vads_ctx_mode	Operating mode.	enum	TEST or PRODUCTION
vads_currency	Code of the currency used for the payment.	n3	E.g.: 978 for euro (EUR)
vads_action_mode	Acquisition mode for payment method data.	enum	INTERACTIVE
vads_identifier	(unique) token associated with a payment method.	ans..50	E.g.: MyToken Note: two possible formats: <ul style="list-style-type: none">• an32: if the identifier is generated by the payment gateway• ans..50: if the identifier is generated by the merchant.
vads_payment_config	Payment type	enum	SINGLE
vads_site_id	Shop ID	n8	E.g.: 12345678
vads_trans_date	Date and time of the payment form in UTC format.	n14	E.g.: 20190501130025
vads_trans_id	Unique ID of a transaction	n6	E.g.: 123456
vads_version	Version of the exchange protocol.	string	V2

2. Add [the fields recommended for increasing chances of frictionless](#) during the payment.
3. Add optional fields according to your requirements (see **Using additional features** chapter of the [Implementation Guide Hosted Payment Page](#)).
4. Compute the value of the **signature** field using all the fields of your form that start with **vads_** (see chapter **Computing the signature** of the [Implementation Guide Hosted Payment Page](#) available on our website).

9.5. Transmitting buyer details

The Merchant can specify the buyer's billing details (e-mail address, title, phone number, etc.). This information will be used to create the invoice.

All the data transmitted via the payment form can be viewed in the transaction details in the Merchant Back Office (**Buyer** tab).

Use optional fields according to your requirements. *These fields will be returned with the response and will include the value transmitted in the form.*

Field name	Description	Format	Value
vads_cust_email	Buyer's e-mail address	ans..150	E.g.: abc@example.com
vads_cust_id	Buyer reference on the merchant website	an..63	E.g.: C2383333540
vads_cust_national_id	National identifier	ans..255	E.g.: 940992310285
vads_cust_title	Buyer's title	an..63	E.g.: M
vads_cust_status	Status	enum	PRIVATE: for a private individual COMPANY: for a company
vads_cust_first_name	First name	ans..63	E.g.: Laurent
vads_cust_last_name	Last name	ans..63	E.g.: Durant
vads_cust_legal_name	Buyer's legal name	ans..100	E.g.: D. & Cie
vads_cust_phone	Phone number	an..32	E.g.: 0467330222
vads_cust_cell_phone	Cell phone number	an..32	E.g.: 06 12 34 56 78
vads_cust_address_number	Street number	ans..64	E.g.: 109
vads_cust_address	Postal address	ans..255	E.g.: Rue de l'Innovation
vads_cust_address2	Address line 2	ans..255	E.g.:
vads_cust_district	District	ans..127	E.g.: Centre ville
vads_cust_zip	Zip code	an..64	E.g.: 31670
vads_cust_city	City	an..128	E.g.: Labège
vads_cust_state	State / Region	ans..127	E.g.: Occitanie
vads_cust_country	Country code in compliance with the ISO 3166 alpha-2 standard	a2	E.g.: "FR" for France, "PF" for French Polynesia, "NC" for New Caledonia, "US" for the United States.

Note: **vads_cust_phone** and **vads_cust_cell_phone** fields accept all formats. Examples:

- 0123456789
- +33123456789
- 0033123456789
- (00.571) 638.14.00
- 40 41 42 42

9.6. Transmitting shipping details

The merchant can transmit the buyer's shipping details (e-mail address, title, phone number etc.).

This information can be found in the transaction details in the Merchant Back Office (**Shipping tab**).

Use optional fields according to your requirements. *These fields will be returned with the response and will include the value transmitted in the form.*

Field name	Description	Format	Value
vads_ship_to_city	City	an..128	E.g.: Bordeaux
vads_ship_to_country	Country code in compliance with the ISO 3166 standard (required for triggering one or more actions if the Shipping country control profile is enabled).	a2	E.g.: FR
vads_ship_to_district	District	ans..127	E.g.: La Bastide
vads_ship_to_first_name	First name	ans..63	E.g.: Albert
vads_ship_to_last_name	Last name	ans..63	E.g.: Durant
vads_ship_to_legal_name	Legal name	an..100	E.g.: D. & Cie
vads_ship_to_phone_num	Phone number	ans..32	E.g.: 0460030288
vads_ship_to_state	State / Region	ans..127	E.g.: Nouvelle Aquitaine
vads_ship_to_status	Allows to specify the type of the shipping address.	enum	PRIVATE : for shipping to a private individual COMPANY : for shipping to a company
vads_ship_to_street_number	Street number	ans..64	E.g.: 2
vads_ship_to_street	Postal address	ans..255	E.g.: Rue Sainte Catherine
vads_ship_to_street2	Address line 2	ans..255	
vads_ship_to_zip	Zip code	an..64	E.g.: 33000

9.7. Transmitting order details

The merchant can indicate in their payment form if they wish to transfer the order details (order reference, description, shopping cart contents, etc.).

This information can be found in the transaction details in the Merchant Back Office (**Shopping cart** tab).

1. Use optional fields according to your requirements. These fields will be returned with the response and will include the value transmitted in the form.

Field name	Description	Format	Value
vads_order_info	Additional order info	ans..255	E.g.: Door code 3125
vads_order_info2	Additional order info	ans..255	E.g.: No elevator
vads_order_info3	Additional order info	ans..255	E.g.: Express
vads_nb_products	Number of items in the cart	n..12	E.g.: 2
vads_product_ext_idN	Product barcode on the merchant website. N corresponds to the index of the item (0 for the first one, 1 for the second one, etc.).		E.g.: vads_product_ext_id0 = "0123654789123654789" vads_product_ext_id1 = "0223654789123654789"
vads_product_labelN	Item name. N corresponds to the index of the item (0 for the first one, 1 for the second one, etc.).	ans..255	E.g.: vads_product_label0 = "Dated 3 days stay" vads_product_label1 = "Private concert"
vads_product_amountN	Item amount expressed in the smallest currency unit. N corresponds to the index of the item (0 for the first one, 1 for the second one, etc.).	n..12	E.g.: vads_product_amount0 = "32150" vads_product_amount1 = "10700"
vads_product_typeN	Item type. N corresponds to the index of the item (0 for the first one, 1 for the second one, etc.).	enum	E.g.: vads_product_type0 = "TRAVEL" vads_product_type1 = "ENTERTAINMENT"
vads_product_refN	Item reference. N corresponds to the index of the item (0 for the first one, 1 for the second one, etc.).	an..64	E.g.: vads_product_ref0 = "1002127784" vads_product_ref1 = "1002127693"
vads_product_qtyN	Item quantity. N corresponds to the index of the item (0 for the first one, 1 for the second one, etc.).	n..12	E.g.: vads_product_qty0 = "1" vads_product_qty1 = "1"

2. Populate the **vads_nb_products** field with the number of items contained in the cart.



This field becomes mandatory for the shopping cart to be taken into account.

When it is populated, the **Shopping cart** tab becomes available in the transaction details in the Merchant Back Office.

However, if the other fields that start with **vads_product_** are not populated, the tab will not include any information. For this reason, when populating the **vads_nb_products** field, it becomes mandatory to populate the other fields that start with **vads_product_**.

3. Populate the **vads_product_amountN** field with the amount for the items in the cart, using the smallest currency unit.

N corresponds to the index of the item (0 for the first one, 1 for the second one, etc.).

4. Populate **vads_product_typeN** with the value corresponding to the item type.

N corresponds to the index of the item (0 for the first one, 1 for the second one, etc.).

Value	Description
FOOD_AND_GROCERY	Food and grocery
AUTOMOTIVE	Cars / Moto
ENTERTAINMENT	Entertainment / Culture
HOME_AND_GARDEN	Home / Gardening
HOME_APPLIANCE	Household appliances
AUCTION_AND_GROUP_BUYING	Auctions / Group purchasing
FLOWERS_AND_GIFTS	Flowers / Presents
COMPUTER_AND_SOFTWARE	Computers / Software
HEALTH_AND_BEAUTY	Health / Beauty
SERVICE_FOR_INDIVIDUAL	Services for individuals
SERVICE_FOR_BUSINESS	Services for companies
SPORTS	Sports
CLOTHING_AND_ACCESSORIES	Clothes / Accessories
TRAVEL	Travel
HOME_AUDIO_PHOTO_VIDEO	Audio / Photo / Video
TELEPHONY	Telephony

5. Populate **vads_product_labelN** with the name of each item contained in the cart.

N corresponds to the index of the item (0 for the first one, 1 for the second one, etc.).

6. Populate **vads_product_qtyN** with the quantity of each item contained in the cart.

N corresponds to the index of the item (0 for the first one, 1 for the second one, etc.).

7. Populate **vads_product_refN** with the reference of each item contained in the cart.

N corresponds to the index of the item (0 for the first one, 1 for the second one, etc.).

8. Check the value of the **vads_amount** field. It must correspond to the total amount of the order.

9.8. Increasing the chances of a frictionless payment



- The use of these fields is optional. In any case, it is the issuing bank that decides if strong authentication must be performed.
- These fields and their values will only be taken into account after 3DS2 is enabled for your MID.
- This is a non-exhaustive list and it will be completed in the future.

Name/Description	Format/Values
vads_cust_address_number Street number - Billing address	Format: ans..64
vads_cust_address2 2nd line of the address - Billing address	Format: ans..255
vads_cust_address 1st line of the address - Billing address	Format: ans..255
vads_cust_cell_phone Buyer's cell phone number	Format: an..32
vads_cust_city City - Billing address	Format: an..128
vads_cust_email Cardholder's e-mail address	Format: ans..150
vads_cust_national_id National identifier. Allows to identify each citizen of a country in a unique way (E.g. CPF/CNPJ in Brazil).	Format: ans..255
vads_cust_phone Shipping buyer's phone number	Format: an..32
vads_cust_state State/Region - Billing address	Format: ans..127
vads_cust_zip Zip code - Billing address	Format: an..64
vads_ship_to_city City - Shipping address	Format: an..128
vads_ship_to_email Shipping e-mail address in case of e-ticket order.	Format: an..128
vads_ship_to_type Transport type New values specific to 3DS2 will be available soon.	Format: enum
vads_ship_to_state State/Region - Shipping address	Format: ans..127
vads_ship_to_street2 2nd line of the address - Shipping address	Format: ans..255
vads_ship_to_street 1st line of the address - Shipping address	Format: ans..255
vads_ship_to_speed Shipping speed New values specific to 3DS2 will be available soon.	Format: enum

Name/Description	Format/Values
vads_ship_to_zip Zip code - Shipping address	Format: ans..64

9.9. Transmitting merchant preferences

The merchant can express their choice concerning strong buyer authentication using the **vads_threeds_mpi** field.

The value transmitted in the payment request has priority over the risk rules potentially defined by the merchant in their Merchant Back Office.

Here is how to use it:

Value	Description
missing or empty or 0	<ul style="list-style-type: none">• 3DS1: Forced 3DS1 authentication.• 3DS2: The choice of the preference is transferred to the card issuer (No Preference).
1	Deprecated.
2	<ul style="list-style-type: none">• 3DS1: Disabled 3DS1 authentication. Deprecated By using this value, you expose yourself to “Soft decline” refusals. If the store does not have the “Selective 3DS1” function, 3DS1 authentication is forced.• 3DS2: Allows to request authentication without interaction (frictionless). <i>Requires the “Frictionless 3DS2” option.</i><ul style="list-style-type: none">• For payments made in euro, if the amount is lower than €30, a request for frictionless is transmitted to the issuer. If the frictionless request is accepted, the transaction does not benefit from liability shift in case of chargeback.• For payments made in euro, if the amount is higher than €30, the value transmitted by the merchant is ignored and the choice of the preference is transferred to the card issuer (No Preference).• For payments made in a currency other than euro, a request for frictionless is transmitted to the issuer. If the frictionless request is accepted, the transaction does not benefit from liability shift in case of chargeback. <p>If the store does not have the “Frictionless 3DS2” option, the choice of the preference is transferred to the card issuer (No Preference).</p>
3	<ul style="list-style-type: none">• 3DS1: Forced 3DS1 authentication.• 3DS2: CHALLENGE REQUESTED: 3DS Requestor Preference. Allows to request strong authentication for the transaction.
4	<ul style="list-style-type: none">• 3DS1: Forced 3DS1 authentication.• 3DS2: CHALLENGE REQUESTED: mandate. Allows to indicate that, due to regulatory reasons, strong authentication is required for the transaction.
5	<ul style="list-style-type: none">• 3DS1: Forced 3DS1 authentication.• 3DS2: NO PREFERENCE: The choice of the preference is transferred to the card issuer. If the issuer decides to perform an authentication without interaction (frictionless), the payment will be guaranteed.

10. SENDING THE PAYMENT REQUEST

The buyer will be able to finalize his/her purchase once he/she is redirected to the payment page.

The buyer's browser must transmit the payment form data.

10.1. Redirecting the buyer to the payment page

The URL of the payment gateway is:

<https://scelliuspaiement.labanquepostale.fr/vads-payment/>

Example of parameters sent to the payment gateway:

```
<form method="POST" action="https://scelliuspaiement.labanquepostale.fr/vads-payment/">
<input type="hidden" name="vads_action_mode" value="INTERACTIVE" />
<input type="hidden" name="vads_amount" value="2990" />
<input type="hidden" name="vads_ctx_mode" value="TEST" />
<input type="hidden" name="vads_currency" value="978" />
<input type="hidden" name="vads_cust_country" value="FR" />
<input type="hidden" name="vads_cust_email" value="me@example.com" />
<input type="hidden" name="vads_cust_first_name" value="John" />
<input type="hidden" name="vads_cust_last_name" value="Doe" />
<input type="hidden" name="vads_cust_phone" value="+33102030405" />
<input type="hidden" name="vads_page_action" value="PAYMENT" />
<input type="hidden" name="vads_payment_cards" value="CB" />
<input type="hidden" name="vads_payment_config" value="SINGLE" />
<input type="hidden" name="vads_site_id" value="12345678" />
<input type="hidden" name="vads_trans_date" value="20190710101407" />
<input type="hidden" name="vads_trans_id" value="362812" />
<input type="hidden" name="vads_version" value="V2" />
<input type="hidden" name="signature" value="NM25DPLKEbtGEHCDHn8MBT4ki6aJI/ODaWhCzCnAfvY=" />
<input type="submit" name="pay" value="Pay"/>
</form>
```

10.2. Processing errors

If the payment gateway detects an error while receiving the form, an error message will appear and the buyer will not be able to proceed to the payment.

In TEST mode

The message indicates the source of the error and provides a link to the error code description to help you fix it.

In PRODUCTION mode

The message simply indicates to the buyer that a technical problem has occurred.

In both cases the merchant receives a notification e-mail.

It contains:

- the source of the error,
- a link to possible causes to facilitate its analysis,
- all the fields of the form.

A description of the error codes with their possible causes is available on our website

<https://scelliuspaiement.labanquepostale.fr/doc/fr-FR/error-code/error-00.html>

11. ANALYZING THE PAYMENT RESULT

The analysis of the payment result is described in the *Hosted Payment Page Implementation Guide* available in our online documentation archive (<https://scelliuspaiement.labanquepostale.fr/doc/fr-FR/>).

This document only describes the steps for processing data relative to the response of a payment made with CB.

11.1. Processing the response data

Here is an example of analysis to guide you through processing the response data.

1. Identify the mode (TEST or PRODUCTION) in which the transaction was created by analyzing the value of the **vads_ctx_mode** field.
2. Identify the order by retrieving the value of the **vads_order_id** field if you have transmitted it to the payment gateway.
Make sure that the order status has not been updated yet.
3. Analyze the operation type transmitted in the **vads_operation_type** field.

Value	Description
DEBIT	Debit transaction.
CREDIT	Refund.
VERIFICATION	Payment method verification.

4. Retrieve the payment result transmitted in the **vads_trans_status** field.

Its value allows you to define the order status.

Value	Description
ABANDONED	Abandoned Payment abandoned by the buyer The transaction has not been created, and therefore cannot be viewed in the Merchant Back Office.
ACCEPTED	Accepted. Status of a VERIFICATION type transaction for which the authorization request or information request has been successfully completed. This status cannot evolve. Transactions with the “ ACCEPTED ” status will never be captured.
AUTHORISED	Waiting for capture The transaction has been accepted and will be automatically captured at the bank on the expected date.
AUTHORISED_TO_VALIDATE	To be validated The transaction, created with manual validation, is authorized. The Merchant must manually validate the transaction in order for it to be captured.

Value	Description
	The transaction can be validated as long as the expiration date of the authorization request has not passed. If the authorization validity period has passed, the payment status changes to EXPIRED . The Expired status is final.
CANCELLED	Canceled The transaction has been canceled by the Merchant.
CAPTURED	Captured The transaction has been captured by the bank.
CAPTURE_FAILED	Capture failed Contact the technical support.
EXPIRED	Expired This status appears in the lifecycle of a payment with deferred capture. The expiry date of the authorization request has passed and the Merchant has not validated the transaction. The account of the cardholder will therefore not be debited.
REFUSED	Refused Transaction is declined.
WAITING_AUTHORISATION	Waiting for authorization The capture delay in the bank exceeds the authorization validity period. An information request (or an authorization request for EUR 1 if the acquirer doesn't support information requests) has been accepted. An authorization request for the total amount will be made one day before the capture date. The transaction capture is automatic.
WAITING_AUTHORISATION_TO_VALIDATE	To be validated and authorized The capture delay in the bank exceeds the authorization validity period. A EUR 1 (or information request about the CB network if the acquirer supports it) authorization has been accepted. The Merchant must manually validate the transaction for the authorization request and the capture to occur.

5. Analyze the **vads_occurrence_type** field to determine if it is a single payment or a payment that is part of a series (subscription or installment payment).

Value	Description
UNITAIRE	Single payment (immediate payment).
RECURRENT_INITIAL	First payment of a series.
RECURRENT_INTERMEDIAIRE	Nth payment of a series.
RECURRENT_FINAL	Last payment of a series.

6. Analyze the **vads_payment_config** field to determine whether it is an **installment payment**.

Field name	Value for an immediate payment	Value for a payment in installments
vads_payment_config	SINGLE	MULTI (the exact syntax is MULTI:first=X;count=Y;period=Z)

For a payment in installments, identify the installment number by retrieving the value of the **vads_sequence_number** field.

Warning: with the application of Soft Decline, the **vads_sequence_number** field no longer allows to easily identify the first installment of a payment in installments. Since the sequence number of the first installment can be different from 1, the sequence number of the second installment will not necessarily be 2.

7. Retrieve the value of the **vads_trans_date** field to identify the payment date.
8. Retrieve the value of the **vads_capture_delay** field to identify the number of days before the capture in the bank.
It will allow you to identify whether the payment is an immediate or a deferred payment.

9. Retrieve the used amount and currency. To do this, retrieve the values of the following fields:

Field name	Description
vads_amount	Payment amount in the smallest currency unit.
vads_currency	Code of the currency used for the payment.
vads_change_rate	Exchange rate used for calculating the effective payment amount (see vads_effective_amount).
vads_effective_amount	Payment amount in the currency used for the capture in the bank.
vads_effective_currency	Currency used for the capture in the bank.

10. Retrieve the value of the **vads_auth_result** field to identify the result of the authorization request.
The list of returned codes is available in the chapter [Analyzing the result of the authorization request](#) on page 53.
11. Retrieve the cardholder authentication result. To do this:

- a. Retrieve the value of the **vads_threeds_enrolled** field to identify the status of the card enrollment.

Value	Description
Empty	Incomplete 3DS authentication process (3DS disabled in the request, the merchant is not enrolled or the payment method is not eligible for 3DS).
Y	Authentication available, cardholder enrolled.
N	Cardholder not enrolled
U	Impossible to identify the cardholder or authentication is not available for the card (e.g. commercial or prepaid cards).

- b. Retrieve the result of cardholder authentication by retrieving the value of the **vads_threeds_status** field.

Value	Description
Empty	Incomplete 3DS authentication (3DS disabled in the request, the cardholder is not enrolled or the payment method is not eligible for 3DS).
Y	Cardholder authentication success.

Value	Description
N	Cardholder authentication error.
U	Authentication impossible.
A	Authentication attempted but not completed.

12. Retrieve the result of fraud checks by identifying the value of the **vads_risk_control** field. This field is sent only if the merchant has:

- subscribed to the "**Risk management**" service,
- enabled at least one verification process in the Merchant Back Office (**Settings > Risk management** menu).

It is populated with the list of values separated by ";" with the following syntax: **vads_risk_control = control1=result1;control2=result2**

The possible values for **control** are:

Value	Description
CARD_FRAUD	Verifies whether the cardholder's card number is on the card greylist.
SUSPECT_COUNTRY	Checks whether the issuing country of the buyer's card is on the list of forbidden countries.
IP_FRAUD	Verifies whether the cardholder's IP address is on the IP greylist.
CREDIT_LIMIT	Checks the purchase frequency and amounts for the same card number, or the maximum amount of an order.
BIN_FRAUD	Checks whether the BIN code of the card is on the BIN code greylist.
ECB	Checks whether the buyer's card is of "e-carte bleue" type.
COMMERCIAL_CARD	Checks whether the buyer's card is a commercial card.
SYSTEMATIC_AUTO	Checks whether the buyer's card is a MAESTRO or VISA ELECTRON card.
INCONSISTENT_COUNTRIES	Checks whether the country of the IP address, the country of the payment card and the buyer's country of residence match.
NON_WARRANTY_PAYMENT	Liability shift.
SUSPECT_IP_COUNTRY	Checks whether the buyer's country, identified by their IP address, is on the list of forbidden countries.

The possible values for **result** are:

Value	Description
OK	OK.
WARNING	Informational control failed.
ERROR	Blocking control failed.

13. Retrieve the card data used for payment.

Field name	Description
vads_acquirer_network	Acquirer network. Populated with CB .

Field name	Description
vads_bank_code	Code of the issuing bank.
vads_bank_label	Name of the issuing bank
vads_bank_product	Product code of the card
vads_brand_management	<p>Indicates:</p> <ul style="list-style-type: none"> whether the buyer used a different brand than the default brand defined by the merchant the brand chosen by the buyer the list of available brands. <p><u>Example of a value:</u></p> <pre>vads_brand_management={"userChoice":true, "brand":"CB", "brandList":"CB VISA"}</pre>
vads_card_brand	Brand of the card used for the payment. E.g.: CB, VISA, VISA_ELECTRON, MASTERCARD, MAESTRO, VPAY
vads_card_country	Country code of the country where the card was issued (alpha ISO 3166-2 code, e.g.: "FR" for France, "PF" for French Polynesia, "NC" for New Caledonia, "US" for the United States).
vads_card_number	Card number used for the payment. The number is masked.
vads_expiry_month	Expiry month between 1 and 12 (e.g.: 3 for March, 10 for October).
vads_expiry_year	Expiry year in 4 digits (e.g.: 2023).

14.Store the value of the **vads_trans_uuid** field. It will allow you to assign unique identification to the transaction if you use the Web Service APIs.

15.Retrieve all the order, buyer and shipping details.

These details will be provided in the response only if they have been transmitted in the payment form.

Their values are identical to the ones submitted in the form.

16.Proceed to order update.

11.2. Analyzing the result of the authorization request

The result of the authorization is specified in the **vads_auth_result** field.

Below are the values that can be returned during a CB payment:

Value	Description	Value	Description
00	Approved or successfully processed transaction	54	Expired card
02	Contact the card issuer	55	Incorrect secret code
03	Invalid acceptor	56	Card absent from the file
04	Keep the card	57	Transaction not allowed for this cardholder
05	Do not honor	58	Transaction not allowed for this cardholder
07	Keep the card, special conditions	59	Suspected fraud
08	Confirm after identification	60	The acceptor of the card must contact the acquirer
12	Incorrect Transaction Code	61	Withdrawal limit exceeded
13	Invalid amount	63	Security rules unfulfilled
14	Invalid cardholder number	65	Exceeded number of withdrawals
15	Unknown issuer	68	Response not received or received too late
17	Canceled by the buyer	75	Number of attempts for entering the secret code has been exceeded
19	Retry later	76	The cardholder is already blocked, the previous record has been saved
20	Incorrect response (error on the domain server)	78	Transaction blocked, first transaction on card not properly unblocked
24	Unsupported file update	80	Contactless payment is not accepted by the issuer
25	Unable to locate the registered elements in the file	81	Unsecured payment is not accepted by the issuer
26	Duplicate registration, the previous record has been replaced	82	CVV, dCVV, iCVV incorrect
27	File update edit error	83	Revocation of all recurring payments for the card
28	Denied access to file	84	R1 - Revocation of recurring payment for the card of a specific Merchant or for the MCC and the card
29	Unable to update	86	6P - Failure of the issuer to verify the data
30	Format error	88	A4 - Misuse of the TRA exemption
31	Unknown acquirer company ID	90	Temporary shutdown
33	Expired card	91	Unable to reach the card issuer
34	Suspected fraud	94	Duplicate transaction
38	Expired card	96	System malfunction
41	Lost card	97	Overall monitoring timeout
43	Stolen card	98	Server not available, new network route requested

Value	Description	Value	Description
46	Customer account closed	99	Initiator domain incident
51	Insufficient balance or exceeded credit limit		

To help you understand the refusal reason, here are some details:

Value	Description
03	Invalid acceptor This code is sent by the card issuer. It refers to a configuration problem on authorization servers. (E.g. closed contract, incorrect MCC declared, etc.). To find out the specific reason of the rejection, the buyer must contact his or her bank.
05	Do not honor This code is sent by the card issuer. This code is used in the following cases: <ul style="list-style-type: none"> Invalid expiry date Invalid CVV Exceeded credit limit Insufficient funds (etc.) To find out the specific reason of the rejection, the buyer must contact his or her bank.
51	Insufficient balance or exceeded credit limit This code is sent by the card issuer. This code appears if the funds on the buyer's account are insufficient for making the purchase. To find out the specific reason of the rejection, the buyer must contact his or her bank.
54	Expiry date exceeded This code is sent by the card issuer. Some issuers return this code when the card is canceled before the expiry date , either by the cardholder or by the bank. There is no specific code for indicating that the card has been canceled.
56	Card absent from the file This code is sent by the card issuer. The entered card number is incorrect or the card number + expiration date combination does not exist.
57	Transaction not allowed for this cardholder This code is sent by the card issuer. This code is used in the following cases: <ul style="list-style-type: none"> The buyer attempts to make an online payment with a cash withdrawal card, The authorized payment limit is exceeded To find out the specific reason of the rejection, the buyer must contact his or her bank.
59	Suspected fraud This code is sent by the card issuer. This code appears when an incorrect CVV code or expiration date has been entered several times. To find out the specific reason of the rejection, the buyer must contact his or her bank.
60	The acceptor of the card must contact the acquirer This code is sent by the card issuer. It refers to a configuration problem on authorization servers. It is used when the merchant ID does not correspond to the used sales channel (e.g.: an e-commerce transaction with a distant sale contract with manual entry of contract data). Contact the customer service to resolve the problem.
81	Unsecured payment is not accepted by the issuer This code is sent by the card issuer. After receiving this code, the payment gateway automatically makes a new payment attempt with 3D Secure authentication, when possible.

12. MANAGING CB TRANSACTIONS FROM THE MERCHANT BACK OFFICE

12.1. Viewing transaction details

1. From the **Management > Transactions** menu, select the tab:
 - **Captured transactions** to list the CB transactions already captured,
 - **Transactions in progress** to list the accepted and declined CB transactions of the day.
2. Double click the desired transaction.

The screenshot shows a web application window titled "Details of a transaction in progress: 146068 (Order reference: qdb-786)". The window has several tabs: "Details", "3D Secure", "Buyer", "Risk assessment", "Advanced risks assessment", and "Event log". The "Details" tab is active, showing the following information:

- Transaction identification**
 - Transaction : 146068
 - Transaction UUID : fce18620b2164be5a66ba681bb7a2fac
 - Order reference : qdb-786
 - Shop : (€)
 - Current amount : EUR 67.97
 - Type : Debit
- Transaction life cycle**
 - Status : Waiting for capture
 - Creation date : 29/04/2020 17:13:46
 - Requested capture date : 29/04/2020 17:13:46
- Payment method**
 - Payment method :
 - Default brand :
 - Card number : 497010XXXXXX0014 (2021/06 - valid)
 - Visa product code :
 - Issuing bank :
 - Type of product : Credit card
- Authorization**
 - Merchant ID (MID) : 50
 - Terminal ID (TID) : 012
 - Authorization return : 0: Transaction was approved or successfully processed
 - Authorization number : 3fed48
 - Authorization date : 29/04/2020 17:13:46

At the bottom of the window, there are buttons: "Validate", "Modify", "Cancel", "Duplicate", "Receipt", and a "Close" button in the bottom right corner.

3. To view the holder authentication details, click on the **3D Secure** tab.

Details of a transaction in progress: 029835 (Order reference: 085-671)

Details
3D Secure
Buyer
Risk assessment
Event log

Summary

Payment method registration to 3D Secure : Enrolled
Buyer authentication : Success
Final status of the authentication process : 3D Secure process completed
Liability shift : Yes

3D Secure v2

DS network : VISA
Bin supported by the protocol : Yes
Protocol supported by the acquirer : Yes
URL of the 3DS Method : https:// /acs/v2/3dsMethod
ACS URL : https:// /acs/v2/creq
Authentication method : Challenge (authentication with cardholder interaction)

Authentication data

Proof of authentication : X*****=
e-commerce indicator : 05
Merchant preference : No preferences

DateEvent

15:01:03Card range present in 3DS2 cache Visa
15:01:033DS Method present for this bin
15:01:07Execution of ACS javascript completed
15:01:08Challenge requested by the ACS
15:01:16Authentication completed with cardholder interaction

Close

Please, See our [3D Secure](#) guide for more information about this tab details.

12.2. Canceling a transaction

The **Cancel** operation is only available for the transactions that have not been captured.

If the acquirer supports it, when the merchant cancels a transaction, the payment gateway automatically sends a reversal request to cancel the authorization request.

If the card issuer accepts the request, the authorization limit of the cardholder's card is restored.

Otherwise, or if the acquirer does not support the reversal, the transaction is canceled and the card limit is restored when the authorization request expires.

If a reversal request is made upon cancellation, it will appear in the transaction details (History tab).

In order to cancel a transaction:

1. Select a transaction with a right-click.
2. Select **Cancel**.
3. Confirm that you wish to definitively cancel the selected transaction.

The transaction status changes to **Canceled**.

Note

*It is possible to **cancel** several transactions at the same time.*

*For this, select all the transactions to be canceled. Press and hold down the **Ctrl key** and **click** for selecting multiple transactions.*

*After the selection, you can click **Cancel** using right-click or via the menu bar and confirm your choice.*

*The transaction statuses will change to **Canceled**.*

12.3. Duplicating a transaction

This function allows to create a new transaction with the exact same characteristics (e.g. card number) as the transaction that was used for duplication.

A duplicated transaction has the same characteristics as all the other transactions, and it can be duplicated itself.

During duplication of a transaction, a new authorization request is made with the card number that corresponds to the original transaction. This transaction does not have a payment guarantee.

The payment receipt will be sent to the buyer if the e-mail is specified for the transaction used for duplication and if the notification rule associated with sending an e-mail to the buyer is active.

Transactions that can be subject to duplication must have one of the following status(es):

- Captured
- Expired
- Canceled
- Refused

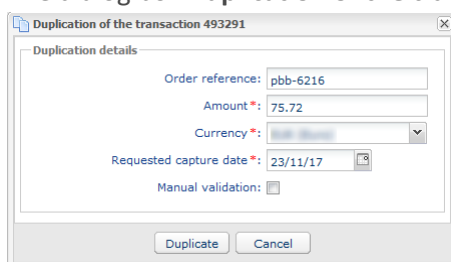
The duplication of refused transactions made with Mastercard cards (Mastercard, Maestro, Mastercard Debit) is forbidden when one of the following reasons is mentioned:

- 04 - Please hold card
- 14 - Invalid cardholder number
- 15 - Unknown card issuer
- 41 - Lost card
- 43 - Stolen card
- 54 - Exp. date of the card passed

To duplicate a transaction:

1. Select the transaction.
2. Click **Duplicate**.

The dialog box **Duplication of the transaction** appears. All of the fields are pre-populated.



You can modify:

- The order reference
- The amount
- The currency

If the selected currency is not supported the following message is displayed: ***Currency not supported by this Merchant ID (MID) and/or shop.***

If the selected currency is supported and multi-currency is possible in your contract, the conversion rate will be applied. The details of the new transaction will be displayed with both currencies: local currency and new currency.

Example

Transaction identification	
Transaction :	908839
Transaction UUID :	7360aa61248c4d5c8801a698be0f723b
UUID Original transaction :	ffa80e59f219484bb6f9db02c498c8dc (Display transaction)
Order reference :	pbb-6216
Shop :	[redacted]
Current amount :	EUR 56.01
Amount in currency :	USD 75.72
Exchange rate :	1.3518
Type :	Debit

- The requested capture date
It can not be earlier than the transaction modification date.
- The validation mode by (un)checking **Manual validation**.

3. Click **Duplicate** to continue or **Cancel** to cancel the duplication.

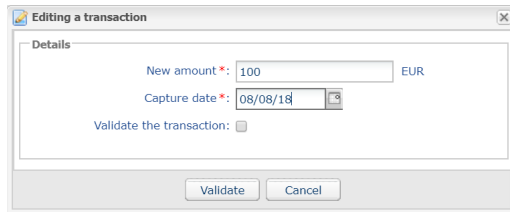
The transaction can be viewed in the **Transactions in progress** tab.

12.4. Modifying a transaction

The **Modify** option is only available as long as the capture date has not been reached.

To modify a transaction:

1. Right-click the transaction
2. Select **Modify**.

A screenshot of a software dialog box titled "Editing a transaction". The dialog has a "Details" section containing two input fields: "New amount" with the value "100" and a currency selector set to "EUR", and "Capture date" with the value "08/08/18" and a calendar icon. Below these fields is a checkbox labeled "Validate the transaction:". At the bottom of the dialog are two buttons: "Validate" and "Cancel".

3. Enter a new amount.

The new amount must be lower than the initial amount.

4. Specify the capture date.

It is also possible to validate a transaction with the **To be validated** or **To be validated and authorized** status by checking **Validate the transaction**.

5. Click **Validate**.

If you wish, you may view the transaction details to see the applied changes (right-click the edited transaction **Display transaction details with**).

12.5. Making a refund

This operation makes it possible to re-credit the buyer's account after a transaction.

The buyer's account is credited with the refunded amount, this same amount is debited from the merchant's account.

The refund is only available on the **captured transactions**.

The refund can be **total** or **partial**.

The merchant is allowed to partially refund many times for the same transaction. The cumulative amount refunded can be up to 300% of the initial transaction amount .

The maximum refund percentage is configured when creating the MID.

On CB network, refund is authorized until the card expiration. Refund is no longer possible after card expiration.

Case of chargebacks: any attempt to refund an unpaid transaction will be rejected.

Case of refund refusal

It is the buyer's bank that objects to the refund request. You must refund your buyer **by another payment method** (cheque, wire transfer, ...).

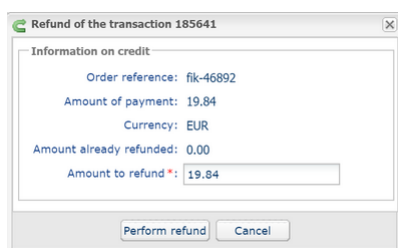
For the CB network, we indicate the code and the refusal reason sent by the buyer's bank. If the refund request is made from the Merchant Back Office, a warning message is also displayed to inform that the buyer's financial institution is the cause of this refusal for its own reason.

For example, if the refund request is made on a blocked card, the code and refusal reason will be "59: suspected fraud" for some acquirers. See the [List of specific return codes](#) for the CB network for more details.

To refund a CB transaction:

1. Go to the **Captured transactions** tab.
2. Right click the transaction to refund.
3. Select **Making a refund** in the context menu.

The **Refund of the transaction** dialog box appears.



Refund of the transaction 185641

Information on credit

Order reference: fik-46892

Amount of payment: 19,84

Currency: EUR

Amount already refunded: 0,00

Amount to refund*: 19,84

Perform refund Cancel

4. Enter the amount to be refunded.
5. Click **Perform refund**.

12.6. Validating a transaction

This operation allows to indicate that the transaction can be captured on the scheduled presentation date.

Only the transactions with the following statuses can be validated:

- **To be validated**
- **To be validated and authorized**

In order to validate a transaction:

1. Click on the tab **Transactions is progress**
2. Select the transaction.
3. Click **Validate**.

Once the transaction has been validated, the status changes to “**Waiting for capture**” or “**Waiting for authorization**” depending on the initial transaction status.

Even if it is not validated before the scheduled capture date, the payment status will remain To be validated until the authorization expires.

In the meantime, you will still be able to validate and/or modify it even if the initial capture date has passed.

Case of installment payments created in manual validation mode:

When a user validates the first installment, a window appears to request confirmation of validation and offer simultaneous validation of all the remaining installments.

Upon each installment validation, and as long as the user has not validated all the remaining installments, this simultaneous validation of remaining installments is suggested.

12.7. Manual reconciliation

This operation allows you to manually reconcile merchant's payments from an account statement.

1. Search for the relevant transaction via the **Captured transactions** tab.
2. Right-click the transaction.
3. Select **Manual reconciliation**.
4. Click **Yes** to confirm the manual reconciliation of the selected transaction.
The **Comment** dialog box appears.
5. Enter a comment for this reconciliation.
6. Click **OK**.

The transaction status changes to **Reconciled**.

12.8. Capturing a transaction

This operation is available during test phase. It is not available in production environment.

The **Capture** option is only available for transactions that have not reached the presentation date.

To manually capture a transaction:

1. Display the tab **Transactions in progress**.
2. Select a transaction with a right-click.
3. Select **Capture manually**.
4. Confirm that you wish to definitively capture the selected transaction.

13. OBTAINING HELP

Looking for help? Check our FAQ on our website

<https://scelliuspaiement.labanquepostale.fr/doc/en-EN/faq/sitemap.html>

For any technical inquiries or if you need any help, contact [technical support](#).

In view of facilitating the processing of your requests, please specify your shop ID (an 8-digit number) in your query.

This information is available in the “registration of your shop” e-mail or in the Merchant Back Office (**Settings > Shop > Configuration**).